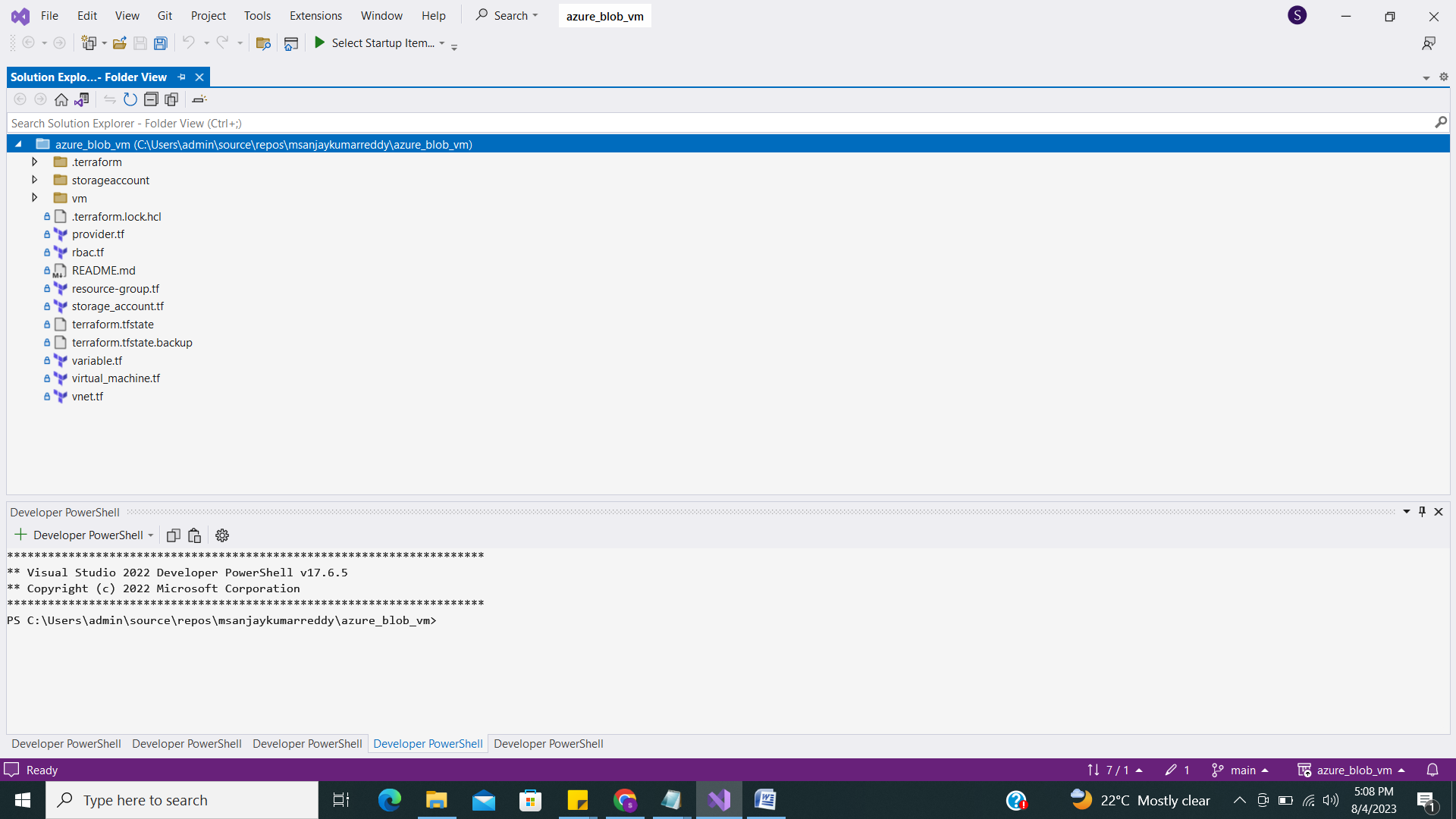
Vm with storage accounts and block devices.  
file structure :  
  
  
  
  
code :  
------------------------------------------------------------------------------------------------------------------------------------------

**Provider.tf**

provider "azurerm" {

features {

}

/\*

client\_id = "129d9ebf-70d0-4caf-b27c-aa35924ce0fe"

client\_secret = "pmv8Q~wVfHrAiQz8r2x.Fb6KuMmzIxvZW6~FEbZc"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

\*/

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

terraform {

required\_providers {

azurerm = {

source = "azurerm"

version = "2.88.1"

}

}

}

**Rbac.tf**

resource "azurerm\_role\_assignment" "storage" {

scope = module.storage\_account.storage\_account\_id

# using azure defined role

role\_definition\_name = "Reader"

principal\_id = module.vm.vm\_pricipal\_id

}

resource "azurerm\_role\_assignment" "container" {

scope = azurerm\_storage\_container.container2.resource\_manager\_id

# using azure defined role

# role\_definition\_name = "Storage Blob Data Contributor"

role\_definition\_name = "Reader"

principal\_id = module.vm.vm\_pricipal\_id

}

**Resourcegroup.tf**  
  
resource "azurerm\_resource\_group" "example" {

name = var.resource\_group\_name

location = "East US"

}  
  
**storageaccounts.tf**  
module "storage\_account" {

source = "./storageaccount"

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

location = azurerm\_resource\_group.example.location

name = "demo12"

white\_list\_ip = ["157.48.67.54"]

whitelist\_subnet\_ids = [azurerm\_subnet.public\_subnet.id]

}

resource "azurerm\_storage\_container" "container" {

name = "demo"

storage\_account\_name = module.storage\_account.storage\_account\_name

container\_access\_type = "private"

depends\_on = [

module.storage\_account

]

}

resource "azurerm\_storage\_container" "container2" {

name = "demo1"

storage\_account\_name = module.storage\_account.storage\_account\_name

container\_access\_type = "private"

depends\_on = [

module.storage\_account

]

}

**Variable.tf**  
variable "resource\_group\_name" {

type = string

default = "qwerty12344321"

}  
  
**virtual\_machine.tf**  
module "vm" {

source = "./vm/"

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

location = azurerm\_resource\_group.example.location

public\_key\_path = "C:/Users/wv3cxq/.ssh/id\_rsa.pub"

name = "demo"

subnet\_id = azurerm\_subnet.public\_subnet.id

}  
  
**vnet.tf**  
resource "azurerm\_virtual\_network" "example" {

name = "example-network"

address\_space = ["10.0.0.0/16"]

location = azurerm\_resource\_group.example.location

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

}

resource "azurerm\_subnet" "public\_subnet" {

name = "public\_subnet"

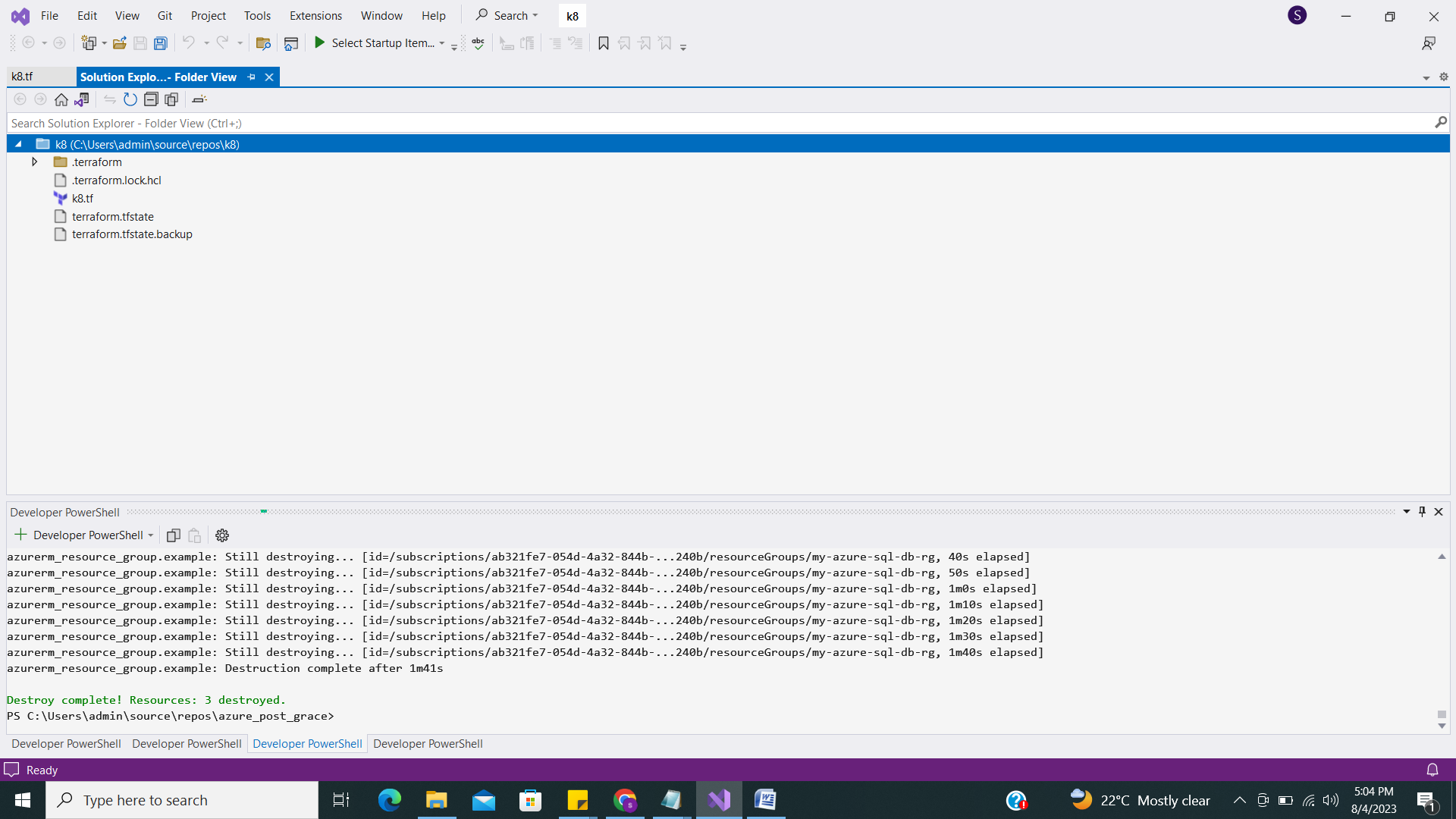
resource\_group\_name = azurerm\_resource\_group.example.name

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

address\_prefixes = ["10.0.1.0/24"]

service\_endpoints = ["Microsoft.Storage"]

}  
------------------------------------------------------------------------------------------------------------------------------------------

k8   
file structure   
  
  
code :  
provider "azurerm" {

features {}

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

resource "azurerm\_resource\_group" "rgkuber" {

name = "kuber"

location = "Central India" # Change this to your desired location

}

resource "azurerm\_kubernetes\_cluster" "example" {

name = "my-k8s-cluster"

location = azurerm\_resource\_group.rgkuber.location

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

dns\_prefix = "my-k8s-cluster" # Replace with your desired DNS prefix

kubernetes\_version = "1.25.5" # Replace with your desired Kubernetes version

default\_node\_pool {

name = "default"

node\_count = 2

vm\_size = "Standard\_DS2\_v2" # Replace with your desired VM size

}

service\_principal {

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda" # Replace with your Service Principal's client\_id

client\_secret = "BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR" # Replace with your Service Principal's client\_secret

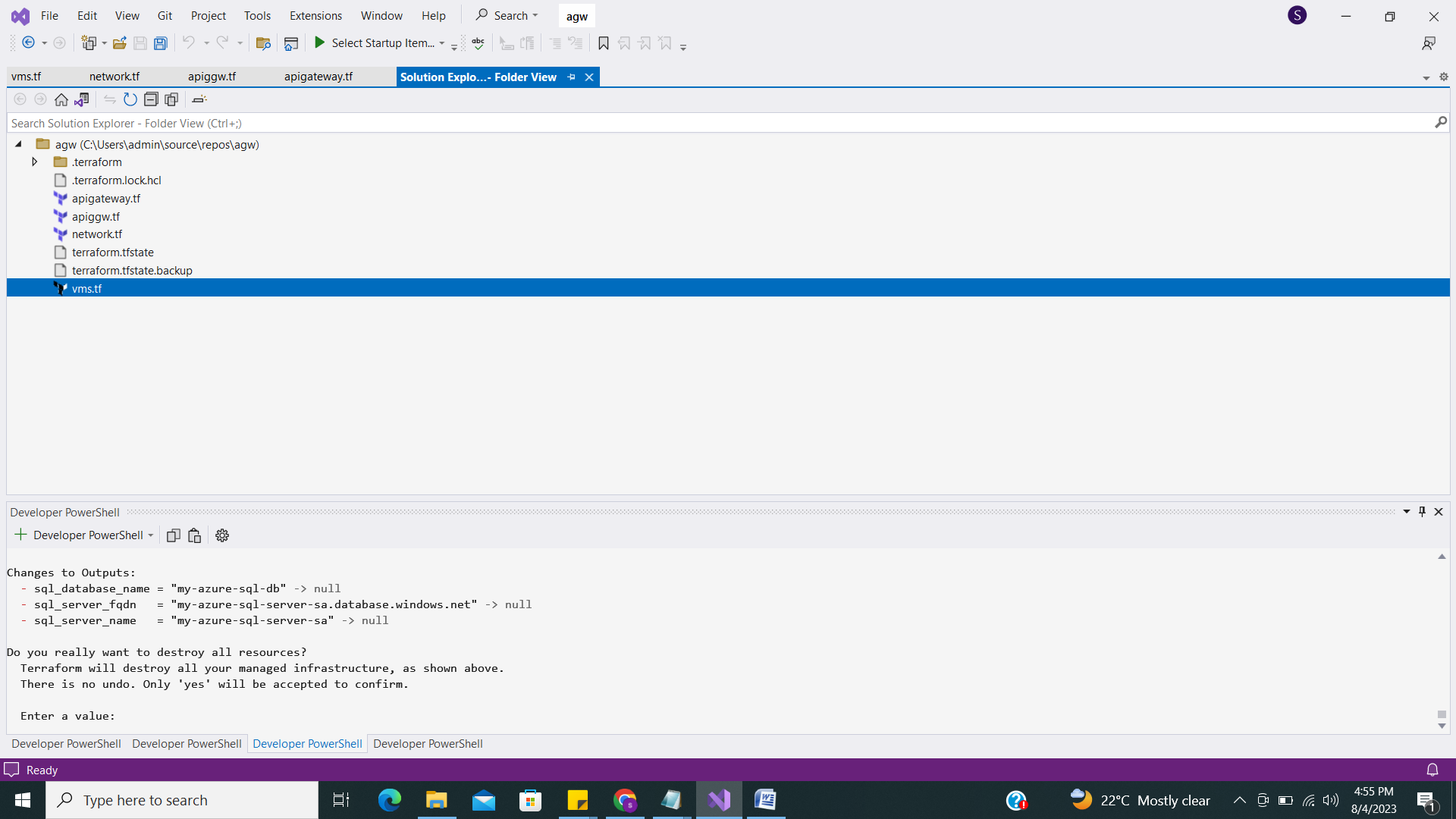
}

tags = {

environment = "dev"

}

}

Application Gateway  
  
file structure:   


Code :   
-----------------------------------------------------------------------------------------------------------------------------------------

Main.tf  
apigatway.tf  
resource "azurerm\_application\_gateway" "example" {

name = "exampleAppGateway"

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

location = azurerm\_resource\_group.example.location

sku {

name = "Standard\_Small"

tier = "Standard"

capacity = 2

}

gateway\_ip\_configuration {

name = "exampleGatewayConfig"

subnet\_id = azurerm\_subnet.example.id

}

frontend\_port {

name = "exampleFrontendPort"

port = 80

}

frontend\_ip\_configuration {

name = "exampleFrontendIP"

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

}

http\_listener {

name = "exampleHTTPListener"

frontend\_ip\_configuration\_name = "exampleFrontendIP"

frontend\_port\_name = "exampleFrontendPort"

protocol = "Http"

}

request\_routing\_rule {

name = "exampleRoutingRule"

rule\_type = "Basic"

http\_listener\_name = azurerm\_application\_gateway.example.http\_listener[0].name

backend\_address\_pool\_name = azurerm\_application\_gateway.example.backend\_address\_pool[0].name

backend\_http\_settings\_name = azurerm\_application\_gateway.example.http\_settings[0].name

}

backend\_address\_pool {

name = "exampleBackendAddressPool"

}

backend\_http\_settings {

name = "exampleHTTPSettings"

cookie\_based\_affinity = "Disabled"

path = "/"

port = 80

protocol = "Http"

request\_timeout = 30

}

http\_request\_routing\_rule {

name = "exampleReqRoutingRule"

rule\_type = "PathBasedRouting"

backend\_address\_pool\_name = azurerm\_application\_gateway.example.backend\_address\_pool[0].name

backend\_http\_settings\_name = azurerm\_application\_gateway.example.http\_settings.name

path\_based\_routing\_configuration {

path\_rule {

name = "examplePathRule"

paths = ["/app1/\*"]

backend\_address\_pool\_name = azurerm\_application\_gateway.example.backend\_address\_pool[0].name

backend\_http\_settings\_name = azurerm\_application\_gateway.example.http\_settings.name

}

}

}

}  
  
**apiggw.tf.**  
  
provider "azurerm" {

features {}

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

resource "azurerm\_resource\_group" "example" {

name = "exampleResourceGroup"

location = "East US"

}

resource "azurerm\_virtual\_network" "example" {

name = "exampleVNET"

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

location = azurerm\_resource\_group.example.location

address\_space = ["10.0.0.0/16"]

}

resource "azurerm\_subnet" "example" {

name = "exampleSubnet"

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

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

address\_prefixes = ["10.0.1.0/24"]

}

resource "azurerm\_public\_ip" "example" {

name = "examplePublicIP"

location = azurerm\_resource\_group.example.location

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

allocation\_method = "Static"

}

**network.tf**  
resource "azurerm\_network\_interface" "music" {

name = "music-nic"

location = azurerm\_resource\_group.example.location

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

ip\_configuration {

name = "music-ip-config"

subnet\_id = azurerm\_subnet.example.id

private\_ip\_address\_allocation = "Dynamic"

}

}

resource "azurerm\_network\_interface" "movie" {

name = "movie-nic"

location = azurerm\_resource\_group.example.location

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

ip\_configuration {

name = "movie-ip-config"

subnet\_id = azurerm\_subnet.example.id

private\_ip\_address\_allocation = "Dynamic"

}

}

**vms.tf**

resource "azurerm\_linux\_virtual\_machine" "music" {

name = "music"

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

location = azurerm\_resource\_group.example.location

size = "Standard\_B1s"

admin\_username = "adminuser"

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

admin\_ssh\_key {

username = "adminuser"

public\_key = file("~/.ssh/id\_rsa.pub") # Replace with the path to your public SSH key

}

os\_disk {

name = "musicOSDisk"

caching = "ReadWrite"

storage\_account\_type = "Standard\_LRS"

}

source\_image\_reference {

publisher = "Canonical"

offer = "UbuntuServer"

sku = "18.04-LTS"

version = "latest"

}

computer\_name = "music-vm"

}

resource "azurerm\_linux\_virtual\_machine" "movie" {

name = "movie"

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

location = azurerm\_resource\_group.example.location

size = "Standard\_B1s"

admin\_username = "adminuser"

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

admin\_ssh\_key {

username = "adminuser"

public\_key = file("~/.ssh/id\_rsa.pub") # Replace with the path to your public SSH key

}

os\_disk {

name = "movieOSDisk"

caching = "ReadWrite"

storage\_account\_type = "Standard\_LRS"

}

source\_image\_reference {

publisher = "Canonical"

offer = "UbuntuServer"

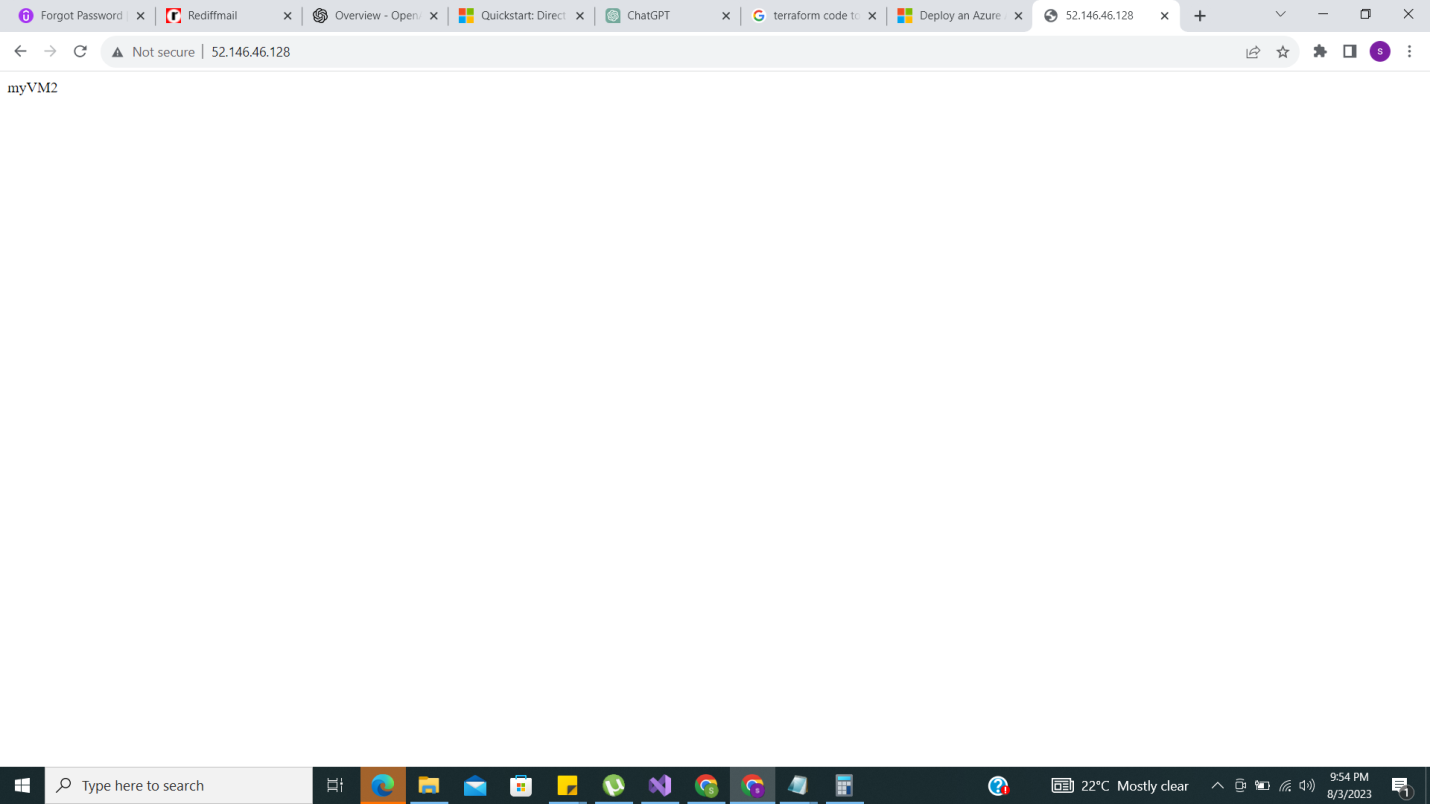
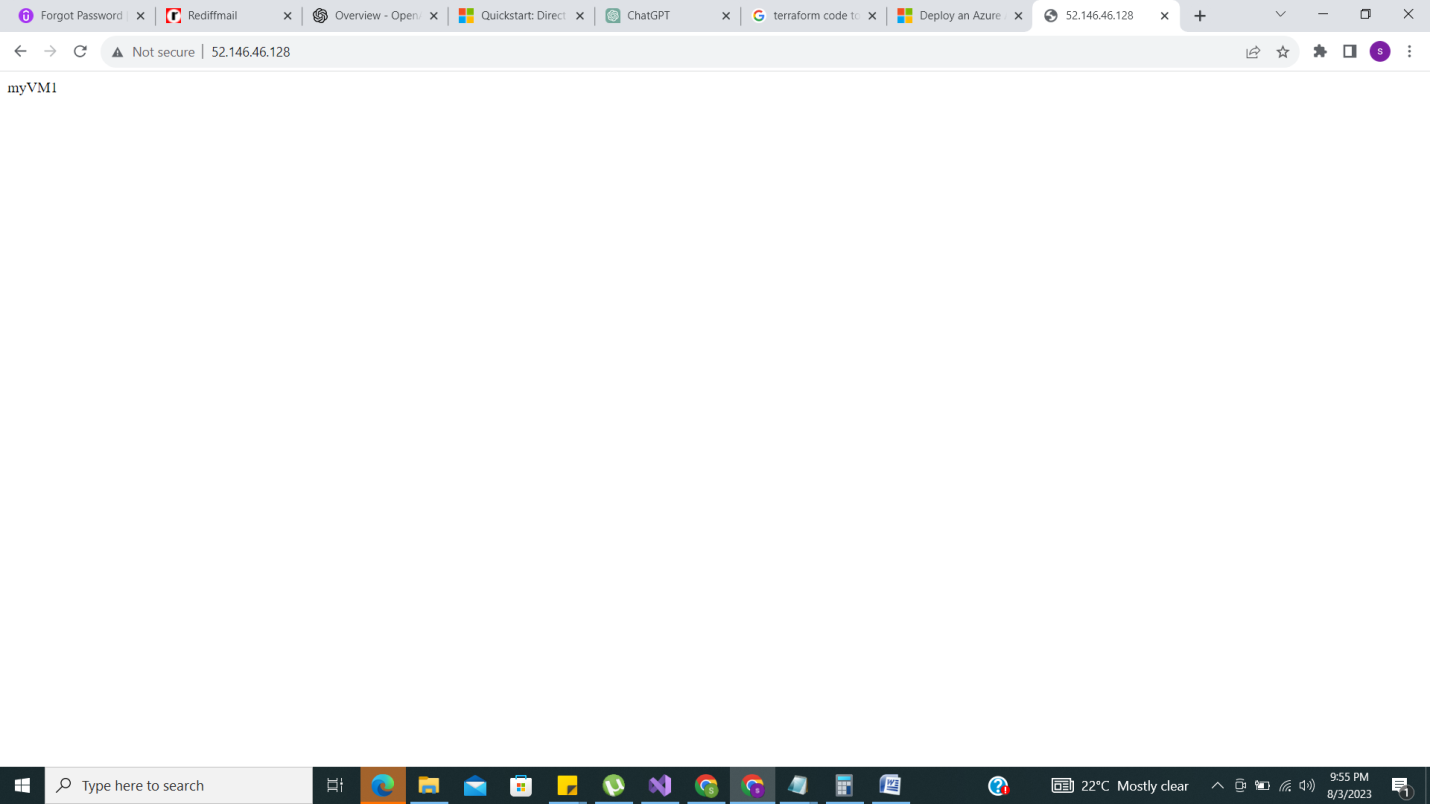
sku = "18.04-LTS"

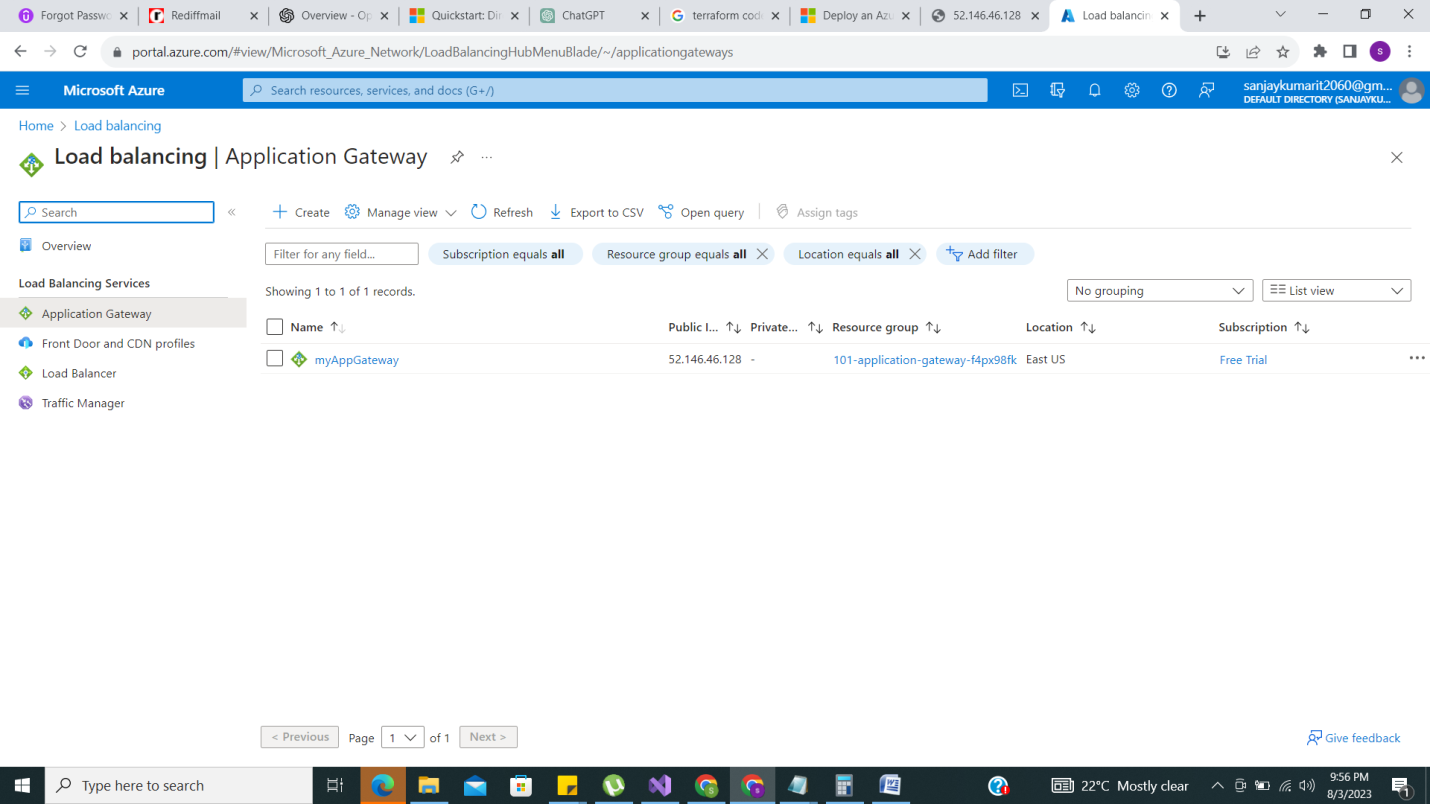
version = "latest"

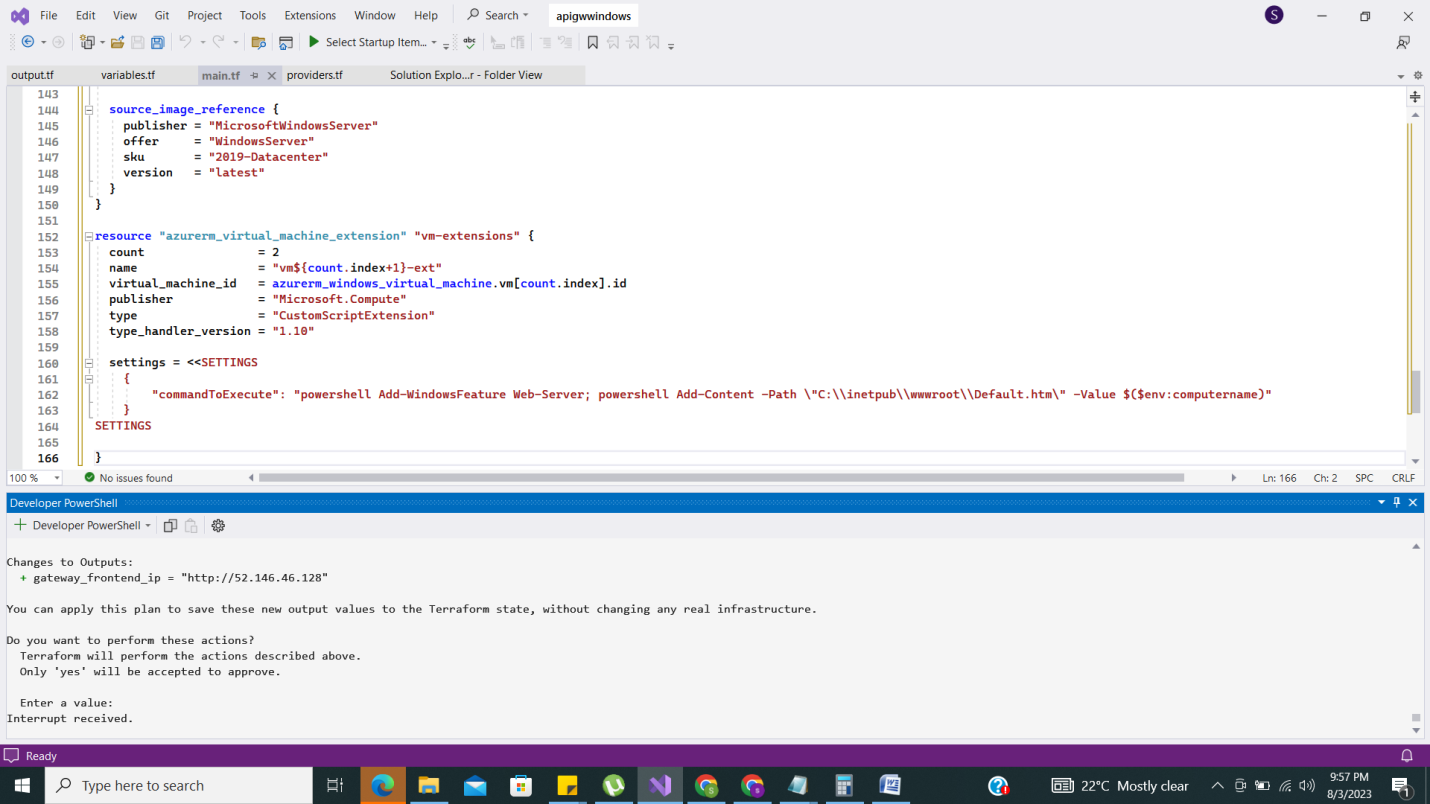
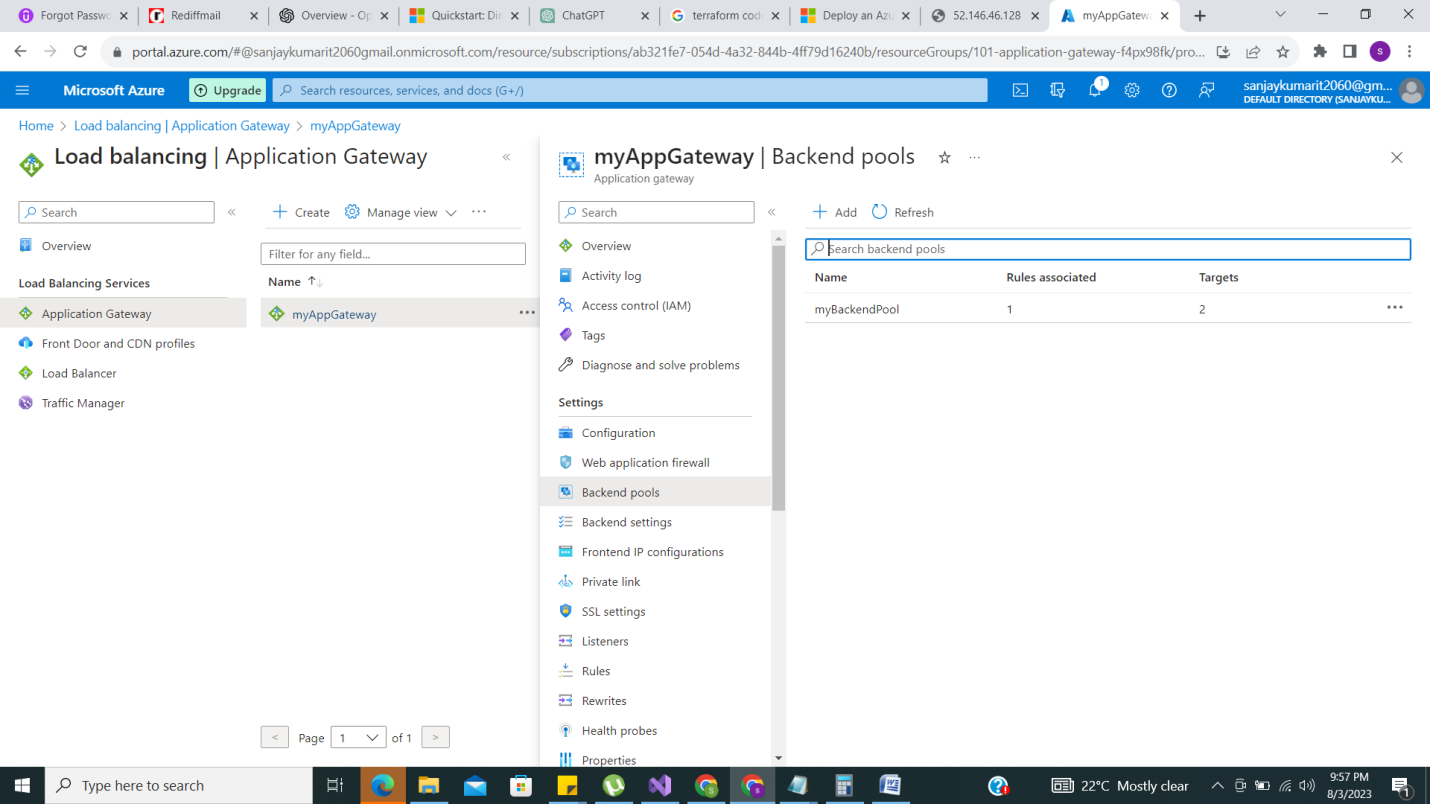
}

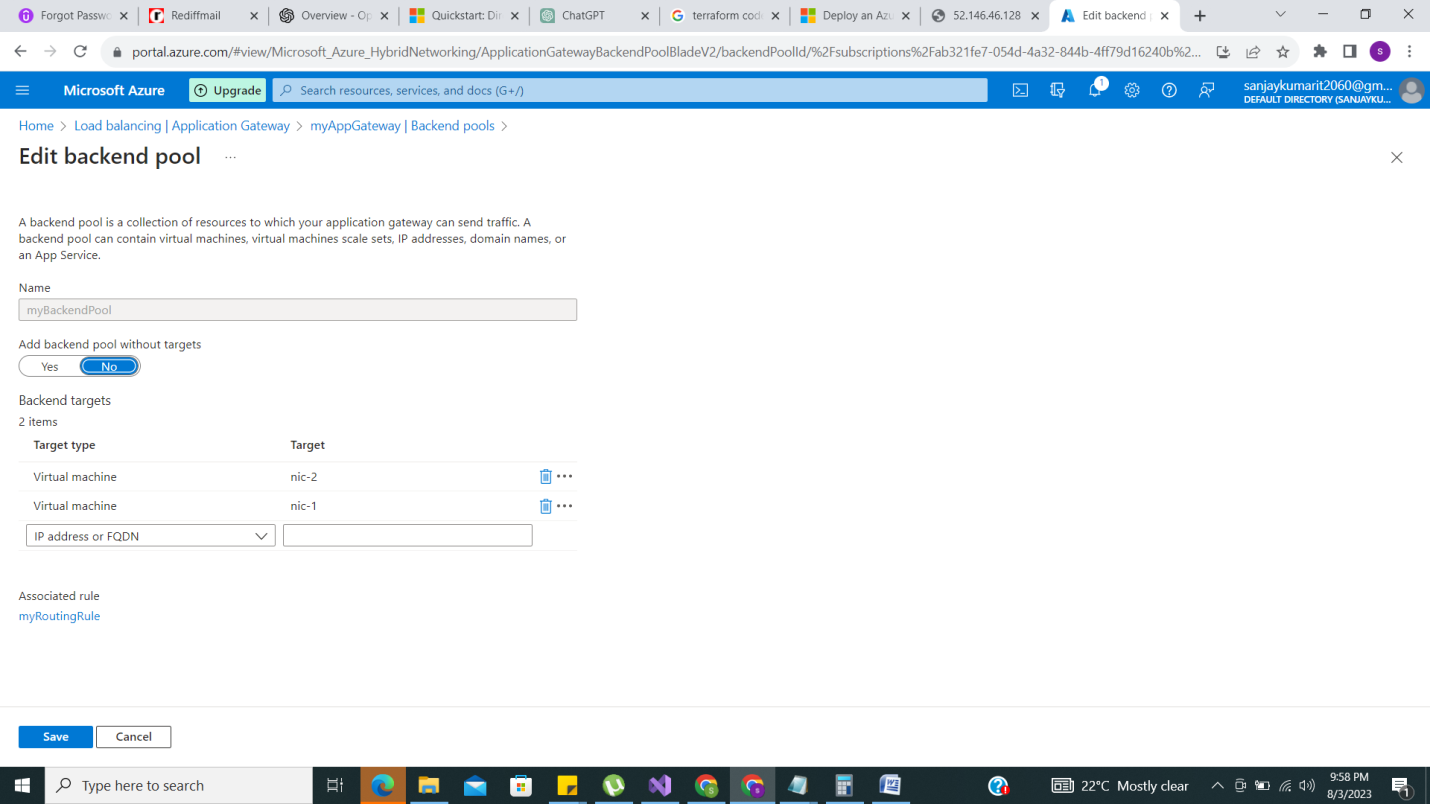
computer\_name = "movie-vm"

}

Output  
  
  
  






azurerm\_sql\_server.

Terraform Code .  
  
------------------------------------------------------------------------------------------------------------------------------------------

provider "azurerm" {

features {}

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

resource "azurerm\_resource\_group" "example" {

name = "my-azure-sql-db-rg"

location = "East US" # Change this to your preferred location

}

resource "azurerm\_mssql\_server" "example" {

name = "my-azure-sql-server-sa"

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

location = azurerm\_resource\_group.example.location

version = "12.0" # Change to the desired SQL Server version

administrator\_login = "sqladmin"

administrator\_login\_password = "P@ssw0rd123!" # Change this to your desired password

/\*

sku {

name = "GP\_Gen5\_2"

tier = "GeneralPurpose"

capacity = 2

family = "Gen5"

}

\*/

}

resource "azurerm\_mssql\_database" "example" {

name = "my-azure-sql-db"

server\_id = azurerm\_mssql\_server.example.id

collation = "SQL\_Latin1\_General\_CP1\_CI\_AS" # Change to your desired collation

}

output "sql\_server\_name" {

value = azurerm\_mssql\_server.example.name

}

output "sql\_server\_fqdn" {

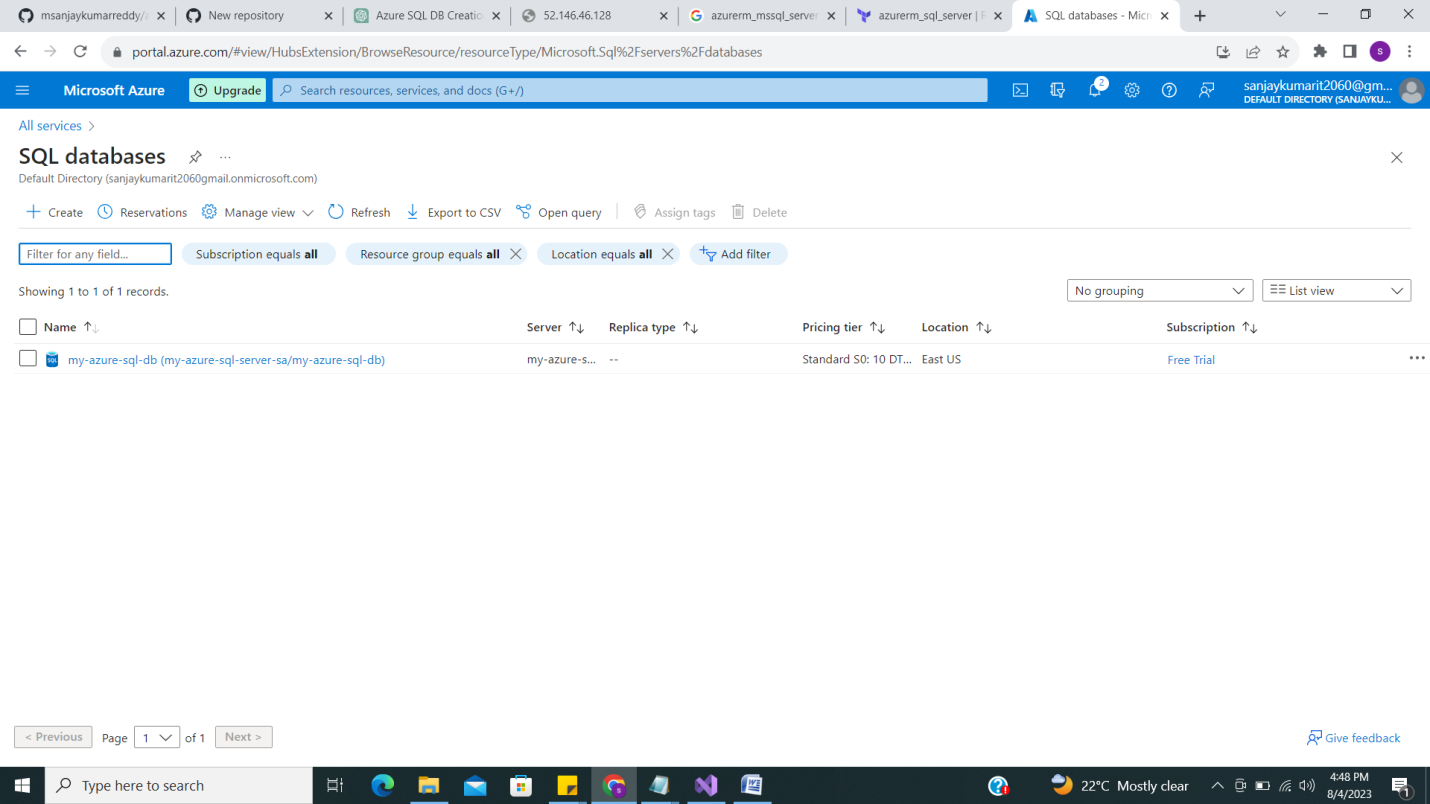
value = azurerm\_mssql\_server.example.fully\_qualified\_domain\_name

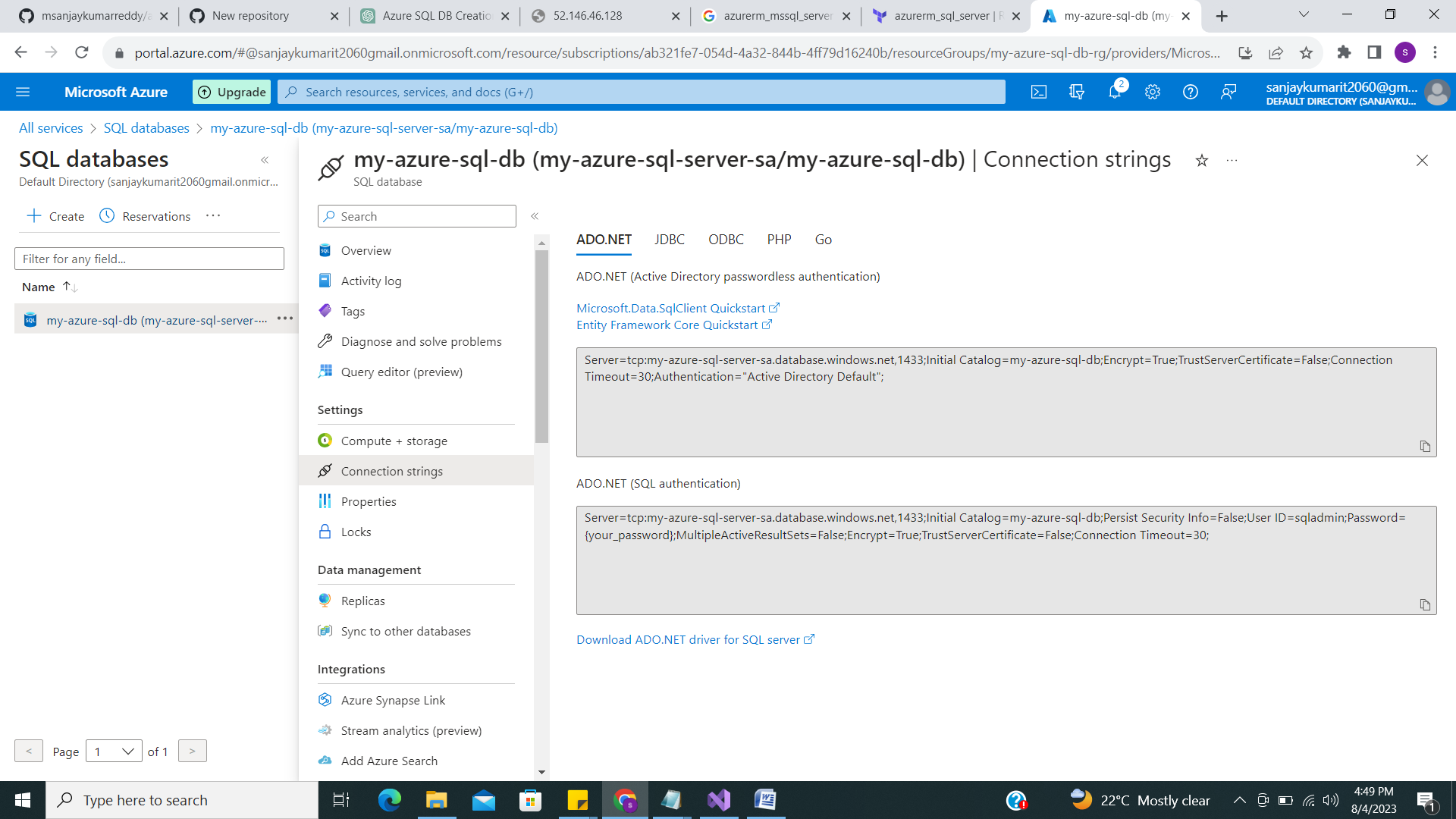
}

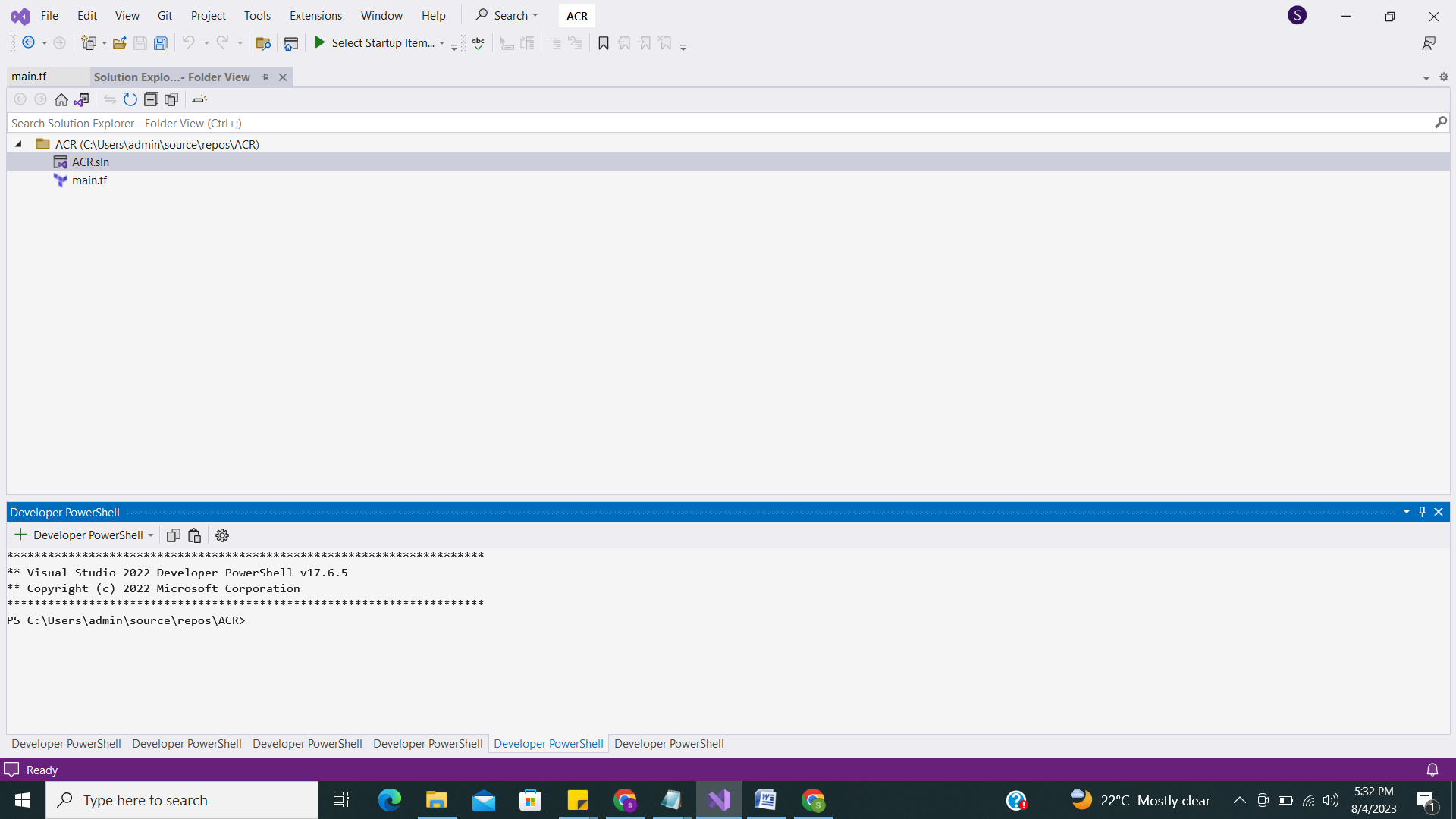
output "sql\_database\_name" {

value = azurerm\_mssql\_database.example.name

}

Screen shots :  
  




**azure container registry :  
  
  
  
code :**provider "azurerm" {

features {}

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

resource "azurerm\_resource\_group" "example" {

name = "my-container-registry-rg"

location = "East US" # Change this to your preferred location

}

resource "azurerm\_container\_registry" "example" {

name = "mycontainerregistrypip"

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

location = azurerm\_resource\_group.example.location

sku = "Basic"

admin\_enabled = true

}

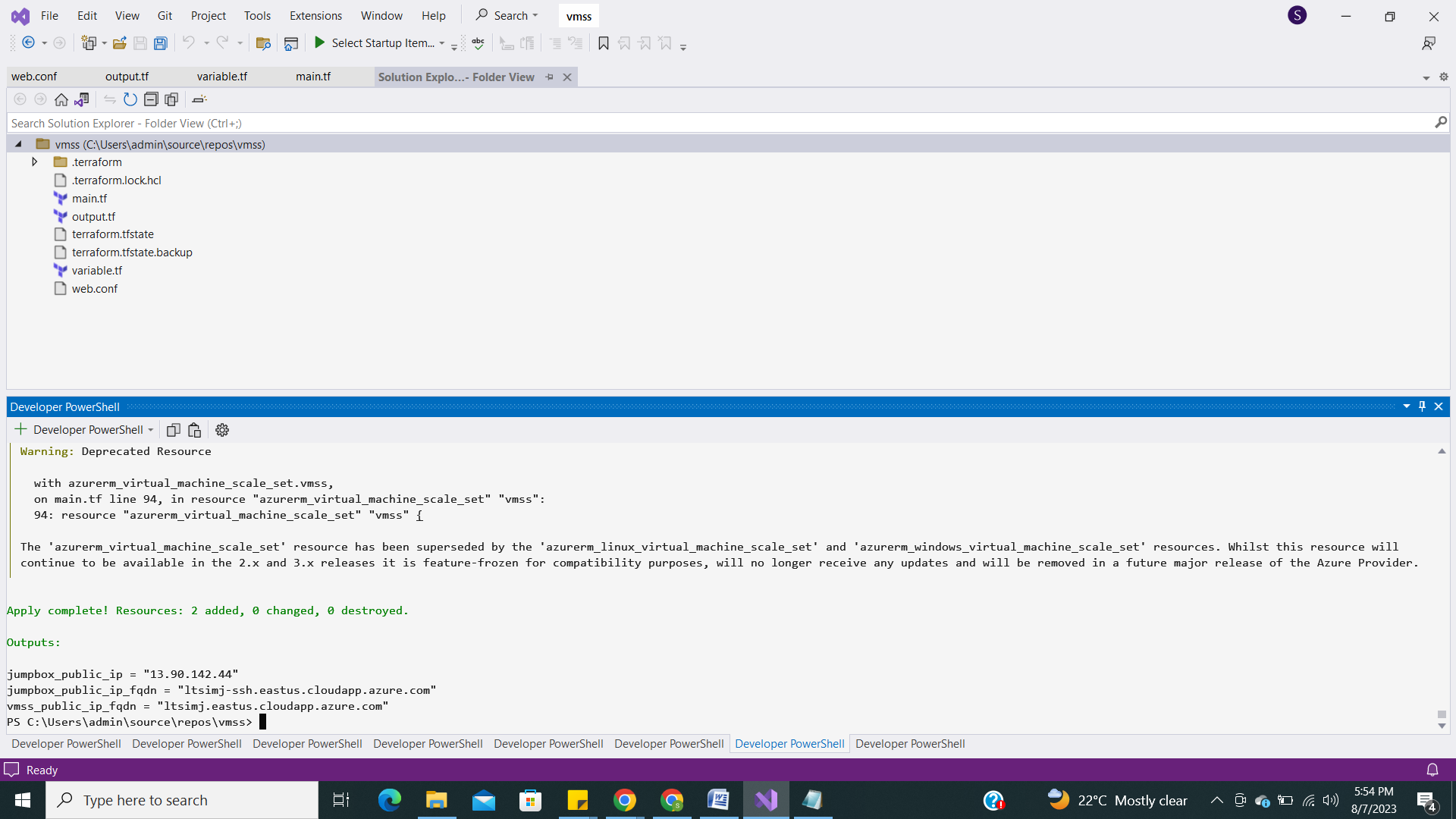
output "container\_registry\_name" {

value = azurerm\_container\_registry.example.name

}

**------------------------------------------------------------------------------------------------------------------------------------------**

**Terraform script VMSS  
  
file stricture.**

****

**Code:   
main.tf .**terraform {

required\_version = ">=0.12"

required\_providers {

azurerm = {

source = "hashicorp/azurerm"

version = "~>2.0"

}

}

}

provider "azurerm" {

features {}

client\_id = "ff574c2d-8d9d-403c-8b8d-b33c2a68fcda"

client\_secret ="BIf8Q~Y6q0ELbHDL0jTjz8pmJhlKySkMPbvVGdbR"

tenant\_id = "bc48d484-77e3-4df8-b4d6-c24f5cde17cb"

subscription\_id = "ab321fe7-054d-4a32-844b-4ff79d16240b"

}

resource "azurerm\_resource\_group" "vmss" {

name = var.resource\_group\_name

location = var.location

tags = var.tags

}

resource "random\_string" "fqdn" {

length = 6

special = false

upper = false

number = false

}

resource "azurerm\_virtual\_network" "vmss" {

name = "vmss-vnet"

address\_space = ["10.0.0.0/16"]

location = var.location

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

tags = var.tags

}

resource "azurerm\_subnet" "vmss" {

name = "vmss-subnet"

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

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

address\_prefixes = ["10.0.2.0/24"]

}

resource "azurerm\_public\_ip" "vmss" {

name = "vmss-public-ip"

location = var.location

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

allocation\_method = "Static"

domain\_name\_label = random\_string.fqdn.result

tags = var.tags

}

resource "azurerm\_lb" "vmss" {

name = "vmss-lb"

location = var.location

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

frontend\_ip\_configuration {

name = "PublicIPAddress"

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

}

tags = var.tags

}

resource "azurerm\_lb\_backend\_address\_pool" "bpepool" {

loadbalancer\_id = azurerm\_lb.vmss.id

name = "BackEndAddressPool"

}

resource "azurerm\_lb\_probe" "vmss" {

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

loadbalancer\_id = azurerm\_lb.vmss.id

name = "ssh-running-probe"

port = var.application\_port

}

resource "azurerm\_lb\_rule" "lbnatrule" {

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

loadbalancer\_id = azurerm\_lb.vmss.id

name = "http"

protocol = "Tcp"

frontend\_port = var.application\_port

backend\_port = var.application\_port

backend\_address\_pool\_id = azurerm\_lb\_backend\_address\_pool.bpepool.id

frontend\_ip\_configuration\_name = "PublicIPAddress"

probe\_id = azurerm\_lb\_probe.vmss.id

}

resource "azurerm\_virtual\_machine\_scale\_set" "vmss" {

name = "vmscaleset"

location = var.location

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

upgrade\_policy\_mode = "Manual"

sku {

name = "Standard\_DS1\_v2"

tier = "Standard"

capacity = 2

}

storage\_profile\_image\_reference {

publisher = "Canonical"

offer = "UbuntuServer"

sku = "16.04-LTS"

version = "latest"

}

storage\_profile\_os\_disk {

name = ""

caching = "ReadWrite"

create\_option = "FromImage"

managed\_disk\_type = "Standard\_LRS"

}

storage\_profile\_data\_disk {

lun = 0

caching = "ReadWrite"

create\_option = "Empty"

disk\_size\_gb = 10

}

os\_profile {

computer\_name\_prefix = "vmlab"

admin\_username = var.admin\_user

admin\_password = var.admin\_password

custom\_data = file("web.conf")

}

os\_profile\_linux\_config {

disable\_password\_authentication = false

}

network\_profile {

name = "terraformnetworkprofile"

primary = true

ip\_configuration {

name = "IPConfiguration"

subnet\_id = azurerm\_subnet.vmss.id

load\_balancer\_backend\_address\_pool\_ids = [azurerm\_lb\_backend\_address\_pool.bpepool.id]

primary = true

}

}

tags = var.tags

}

resource "azurerm\_public\_ip" "jumpbox" {

name = "jumpbox-public-ip"

location = var.location

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

allocation\_method = "Static"

domain\_name\_label = "${random\_string.fqdn.result}-ssh"

tags = var.tags

}

resource "azurerm\_network\_interface" "jumpbox" {

name = "jumpbox-nic"

location = var.location

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

ip\_configuration {

name = "IPConfiguration"

subnet\_id = azurerm\_subnet.vmss.id

private\_ip\_address\_allocation = "dynamic"

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

}

tags = var.tags

}

resource "azurerm\_virtual\_machine" "jumpbox" {

name = "jumpbox"

location = var.location

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

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

vm\_size = "Standard\_DS1\_v2"

storage\_image\_reference {

publisher = "Canonical"

offer = "UbuntuServer"

sku = "16.04-LTS"

version = "latest"

}

storage\_os\_disk {

name = "jumpbox-osdisk"

caching = "ReadWrite"

create\_option = "FromImage"

managed\_disk\_type = "Standard\_LRS"

}

os\_profile {

computer\_name = "jumpbox"

admin\_username = var.admin\_user

admin\_password = var.admin\_password

}

os\_profile\_linux\_config {

disable\_password\_authentication = false

}

tags = var.tags

}

**Output.tf.**output "vmss\_public\_ip\_fqdn" {

value = azurerm\_public\_ip.vmss.fqdn

}

output "jumpbox\_public\_ip\_fqdn" {

value = azurerm\_public\_ip.jumpbox.fqdn

}

output "jumpbox\_public\_ip" {

value = azurerm\_public\_ip.jumpbox.ip\_address

}  
  
variable.tf  
variable "resource\_group\_name" {

description = "Name of the resource group in which the resources will be created"

default = "myResourceGroup"

}

variable "location" {

default = "eastus"

description = "Location where resources will be created"

}

variable "tags" {

description = "Map of the tags to use for the resources that are deployed"

type = map(string)

default = {

environment = "codelab"

}

}

variable "application\_port" {

description = "Port that you want to expose to the external load balancer"

default = 80

}

variable "admin\_user" {

description = "User name to use as the admin account on the VMs that will be part of the VM scale set"

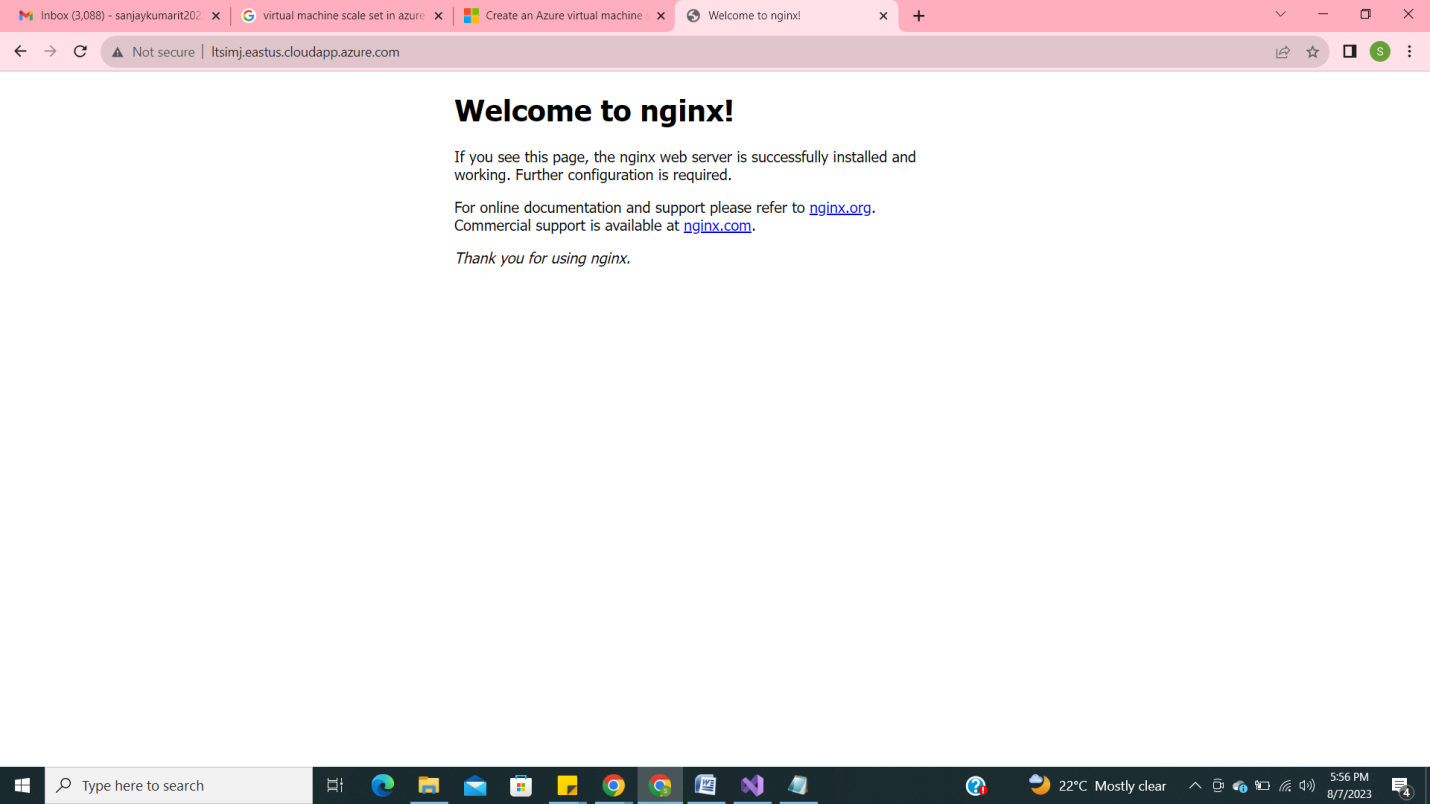
default = "azureuser"

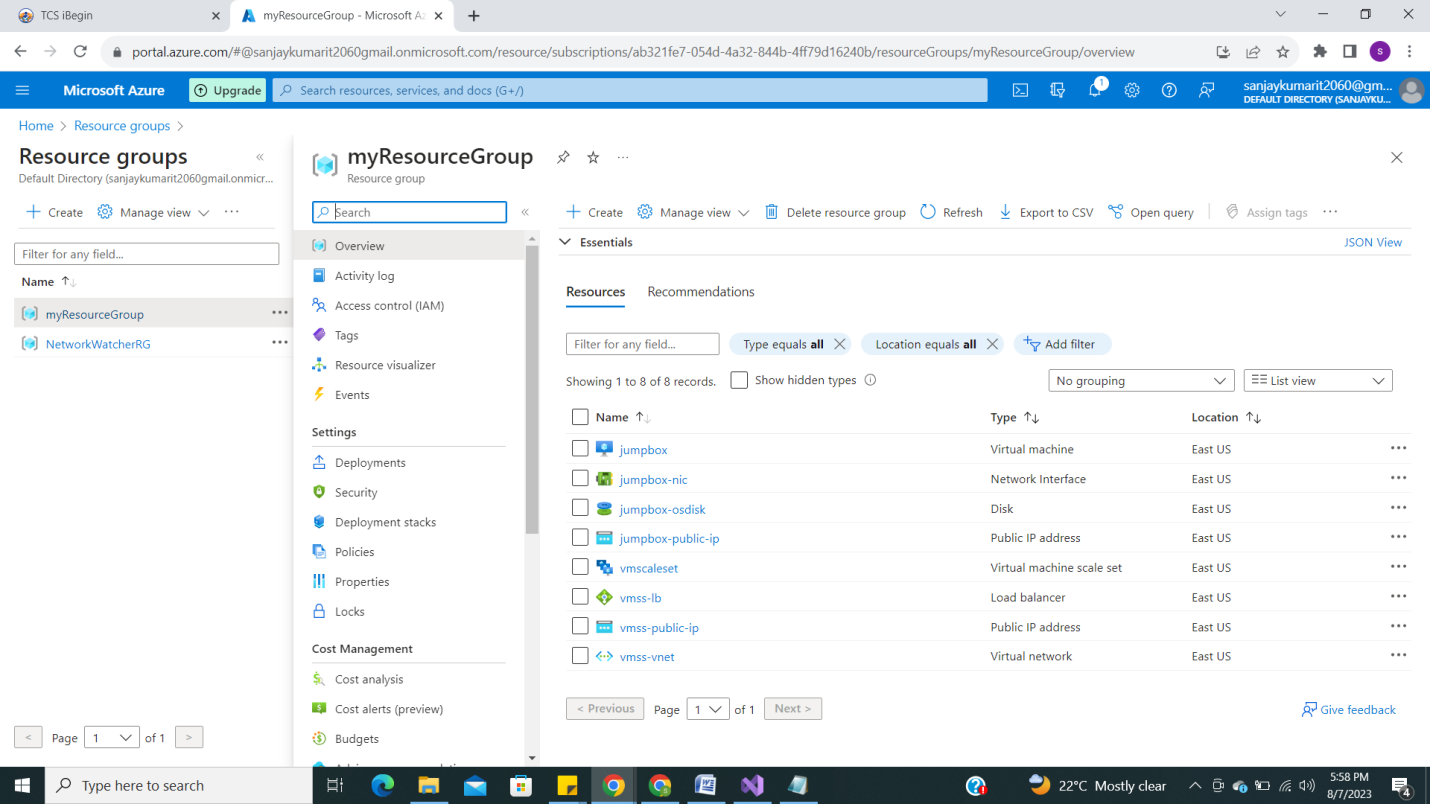
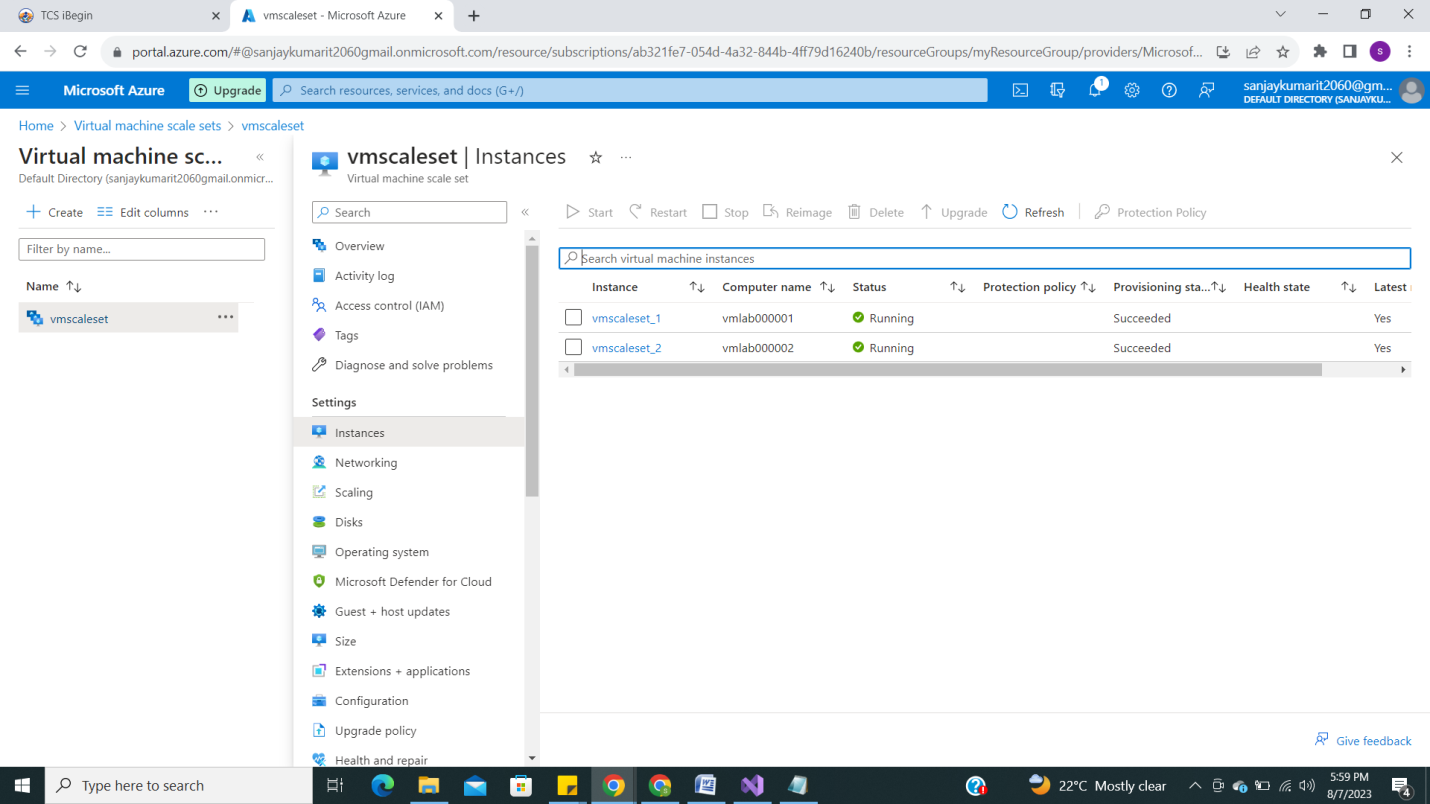
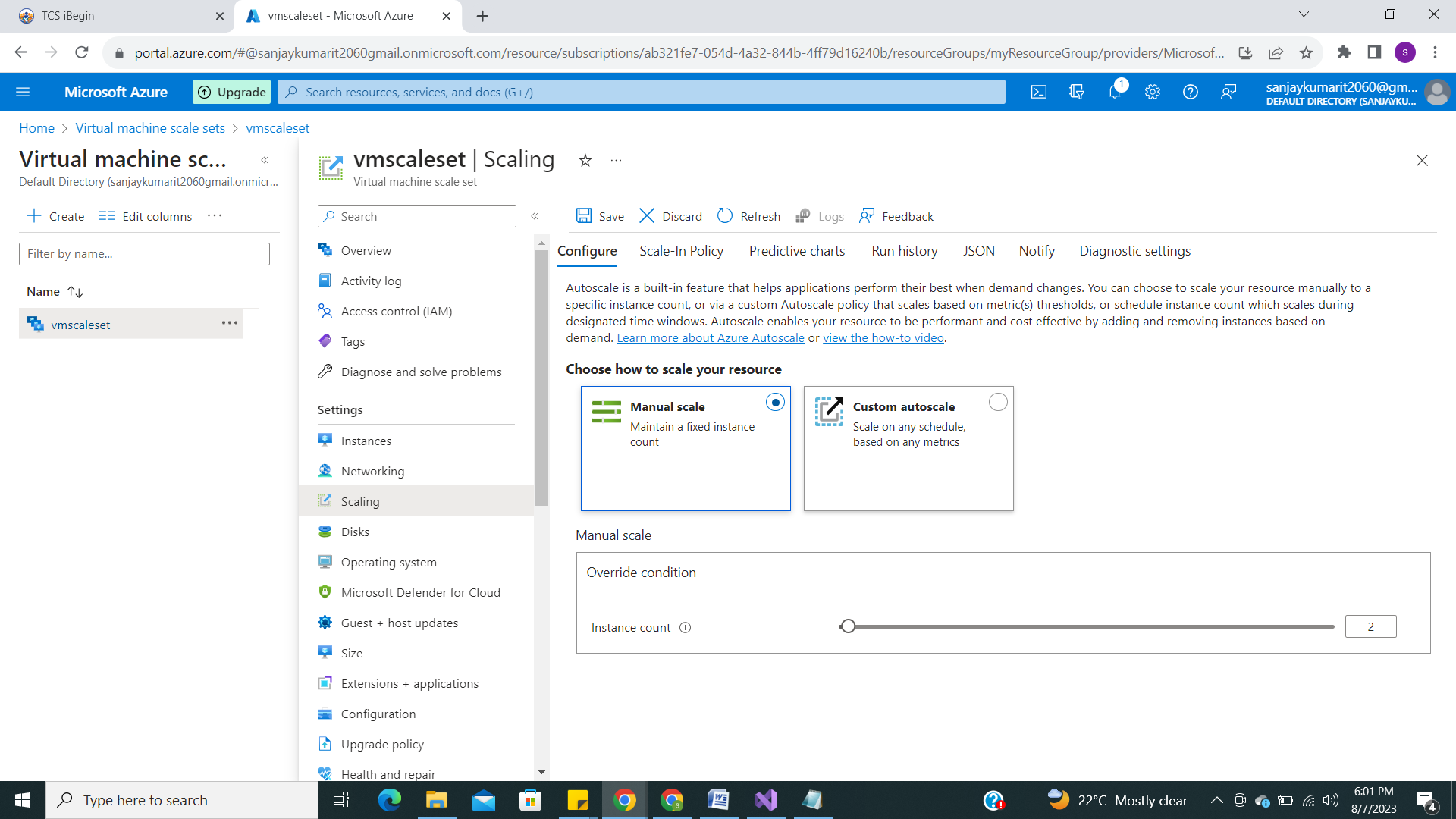
}

variable "admin\_password" {

description = "Default password for admin account"

}  
  
---------------------------------------------------------------------------------

Screen shots.  
  
****

**  
  
  
  
**

**(Key clock),Azure Cache For Redis.**