1. First we started with the creation **of version.tf** which included **two providers azurerm and random.**
2. terraform {
3. required\_providers {
4. azurerm = {
5. source  = "hashicorp/azurerm"
6. version = "=3.0.0"
7. }
8. random = {
9. source = "hashicorp/random"
10. }
11. }
12. }
13. # Configure the Microsoft Azure Provider
14. provider "azurerm" {
15. features {}
16. }
17. Then we created the **generic input variable.tf** which included the default value for 4 variables namely 1)business division 2) environment 3)rg name 4) rg location
18. Created the **random.tf** to generate the random string with certain rule as in length , case, numeric specified
19. resource "random\_string" "myrandom" {
20. length = 6
21. upper = false
22. special = false
23. number = false
24. }
25. Then we created the **local.tf** which includes the creation of value that r repetataive and need some concatenation usingthe default values declared in generic input\_variable.tf ex- rg\_name\_prefic, common tags

#define the local value in terraform

locals {

  owners = var.business\_division

  environment = var.environment

  resource\_name\_prefix = "${var.business\_division}-${var.environment}"

                          #sap                         #dev

  ####sap-dev

  common\_tags = {

      creator = local.owners

      #key = value how do i get the value(in value we have define local.owner=which container var.devision it is coming from generic input variabel file)

      environment = local.environment

  }

}

1. **Created rg** with name location and tag specidied using the values declared in variable.tf, input.tf ex- rg name is concatenation of business dividion, environment, rg default value and random string [azurerm\_resource\_group | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/resource_group)
2. We next **created vnet\_input variable.tf** which included default values which can be used further while vnet and subnet creation
3. #here we are going to define varaible for vnet nsg and subnet
4. variable "vnet\_name" {
5. type = string
6. default = "vnet-default"
7. }
8. variable "vnet\_address\_space" {
9. type = list(string)
10. default = ["10.0.0.0/16"]
11. }
12. #web subnet name and address
13. variable "web\_subnet\_name" {
14. type = string
15. default = "websubnet"
16. }
17. variable "web\_subnet\_adress" {
18. type = list(string)
19. default = ["10.0.1.0/24"]
20. }

Exp-name and address space for vnet and subnet that r required

1. We created the **vnet and sunbet** [azurerm\_virtual\_network | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/virtual_network) , [azurerm\_subnet | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/subnet)
2. We also **created nsg** which serve as virtual firewall in azure and then **associated nsg with subnet. By declaring subnet\_id and nsg\_id arguments** [azurerm\_network\_security\_group | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/network_security_group), [azurerm\_network\_security\_group | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/network_security_group)
3. resource "azurerm\_subnet" "websubnet" {
4. name = "${azurerm\_virtual\_network.vnet.name}-${var.web\_subnet\_name}"
5. #name of the subnet=vnetname-variable define in subnet
6. resource\_group\_name = azurerm\_resource\_group.rg.name #coming from tfstate file
7. virtual\_network\_name = azurerm\_virtual\_network.vnet.name #tfstate name
8. address\_prefixes = var.web\_subnet\_adress
9. }
10. resource "azurerm\_network\_security\_group" "web\_subnet\_nsg" {
11. name = "${azurerm\_subnet.websubnet.name}-nsg"
12. location = azurerm\_resource\_group.rg.location
13. resource\_group\_name = azurerm\_resource\_group.rg.name
14. }
15. ####asociate nsg and subnet
16. resource "azurerm\_subnet\_network\_security\_group\_association" "web\_subnet\_nsg\_associate" {
17. subnet\_id                 = azurerm\_subnet.websubnet.id
18. network\_security\_group\_id = azurerm\_network\_security\_group.web\_subnet\_nsg.id
19. }
20. #locaL BLOCK FOR NSG
21. locals {
22. web\_inbound\_ports\_maps = {
23. "100" : "80",
24. "110" : "443",
25. "120" : "22"
26. }
27. }
28. # nsg inbound rule
29. resource "azurerm\_network\_security\_rule" "web\_nsg\_rule\_inbound" {
30. for\_each = local.web\_inbound\_ports\_maps
31. name                        = "Rule-Port-${each.value}" #Rule-port-80
32. priority                    = each.key #100
33. direction                   = "Outbound"
34. access                      = "Allow"
35. protocol                    = "Tcp"
36. source\_port\_range           = "\*"
37. destination\_port\_range      = each.value #80
38. source\_address\_prefix       = "\*"
39. destination\_address\_prefix  = "\*"
40. resource\_group\_name         = azurerm\_resource\_group.rg.name
41. network\_security\_group\_name = azurerm\_network\_security\_group.web\_subnet\_nsg.name
42. }
43. As initially we need to open the firewall to allow the incoming request--- **do this by creating network security rule for the ports required with the priority mentioned**

We need 3 ports with diff priority hence declared locals for it, by creating key value pair **of priority:port number**

Instead of creating the nsg rule for each port we used **the for\_each** to iterate and make use of local map that we created [azurerm\_network\_security\_rule | Resources | hashicorp/azurerm | Terraform Registry](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/network_security_rule)

1. We then printed desired output to console to avoid it checking in portal.
2. #output the vnet
3. output "virtual\_network\_name" {
4. description = "Virtual Network Name"
5. value = azurerm\_virtual\_network.vnet.name
6. }
7. output "virtual\_network\_id" {
8. description = "Virtual Network id"
9. value = azurerm\_virtual\_network.vnet.id
10. }
11. output "web\_subnet\_name" {
12. description = "web subnet name"
13. value = azurerm\_subnet.websubnet.name
14. }
15. output "web\_subnet\_id" {
16. description = "web subnet name"
17. value = azurerm\_subnet.websubnet.id
18. }
19. output "web\_subnet\_nsg\_name" {
20. description = "web subnet nsg name"
21. value = azurerm\_network\_security\_group.web\_subnet\_nsg.name
22. }
23. output "web\_subnet\_nsg\_id" {
24. description = "web subnet nsg id"
25. value = azurerm\_network\_security\_group.web\_subnet\_nsg.id
26. }