

Lab 02 (Modules 01, 02, and 03)

Create, modify, and destroy resources.



SZABIST

Name	Aliyan
Registration Id	2112104
Section	BSCS 8 th A
Course	Introduction to DevOps

Instructions:

1. Log in to Azure Portal with your credentials.
2. Paste all screenshots (highlighted in red) in a single Word document in the correct order.
3. Name the document as YourName-lab02.

Lab Objective:

- Create, modify, and destroy resources.

Lab Description:

- Create a single Terraform script called lab02.tf containing the following:
 - Provider and Terraform blocks.
 - Code to build the required infrastructure.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Open a Command Prompt or PowerShell window.
 2. Create a directory called lab02 in your home directory.
 3. Change into the lab02 directory.
 4. Create an empty file called lab02.tf.
-

Part 2: Codify the Following in a Single Terraform Script

Make sure to enclose values within double quotation marks.

5. Open lab02.tf in a text editor (e.g., Notepad or Visual Studio Code) and define resource blocks as follows:

Define a resource group called lab02-rg using azurerm_resource_group.

Define a virtual network called lab02-vnet using azurerm_virtual_network.

Add a subnet to the virtual network called lab02-subnet1 using azurerm_subnet.

Define a network security group called lab02-nsg1 with an inbound allow TCP rule for port 22 called rule1 with priority 100 using azurerm_network_security_group.

Attach the network security group lab02-nsg1 to lab02-subnet1 using azurerm_subnet_network_security_group_association.

SCREENSHOT of the Entire Terraform script (capture the full content of lab02.tf in your editor).

File Edit View

```
# Provider Block
provider "azurerm" {
  features {}

  subscription_id          = "ea6fc9c7-fe32-4565-b42a-e336c72040d9"
  resource_provider_registrations = "none"
}

# Random number to append to container name (optional but good practice)
resource "random_integer" "rand" {
  min = 10000
  max = 99999
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
  name      = "lab02-rg"
  location  = "East US"
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
  name          = "lab02-vnet"
  location      = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
  address_space   = ["10.0.0.0/16"]
}

# Subnet
resource "azurerm_subnet" "lab02_subnet1" {
  name          = "lab02-subnet1"
  resource_group_name = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes = ["10.0.1.0/24"]
}

# Network Security Group
resource "azurerm_network_security_group" "lab02_nsg1" {
  name      = "lab02-nsg1"
  location  = azurerm_resource_group.lab02_rg.location
```

File Edit View

```
location          = azurerm_resource_group.lab02_rg.location
resource_group_name = azurerm_resource_group.lab02_rg.name

security_rule {
    name          = "SSH"
    priority      = 100
    direction     = "Inbound"
    access         = "Allow"
    protocol       = "Tcp"
    source_port_range = "*"
    destination_port_range = "22"
    source_address_prefix = "*"
    destination_address_prefix = "*"
}
}

# Attach NSG to Subnet
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc" {
    subnet_id          = azurerm_subnet.lab02_subnet1.id
    network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}

# Container Group (using Microsoft's public image)
resource "azurerm_container_group" "lab02_container" {
    name          = "lab02-container-${random_integer.rand.result}"
    location      = azurerm_resource_group.lab02_rg.location
    resource_group_name = azurerm_resource_group.lab02_rg.name
    os_type        = "Linux"
    ip_address_type = "Public"
    dns_name_label = "lab02-container-${random_integer.rand.result}"

    container {
        name      = "helloworld"
        image    = "mcr.microsoft.com/azuredocs/aci-helloworld" #  UPDATED IMAGE
        cpu      = 1
        memory   = 1.5
        ports {
            port      = 80
            protocol = "TCP"
        }
    }
}
```

```
container {
    name      = "helloworld"
    image    = "mcr.microsoft.com/azuredocs/aci-helloworld" #  UPDATED IMAGE
    cpu      = 1
    memory   = 1.5
    ports {
        port      = 80
        protocol = "TCP"
    }
}

tags = {
    environment = "lab"
}
}
```

Part 3: Initialize Terraform

6. Initialize Terraform and download plug-ins as required:

```
terraform init
```

7. View the content of the terraform.tfstate file:

```
type terraform.tfstate
```

SCREENSHOT (capture the Command Prompt or PowerShell output).

```
C:\Users\SCS\lab02>terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/azurerm from the dependency lock file
- Reusing previous version of hashicorp/random from the dependency lock file
- Using previously-installed hashicorp/azurerm v4.26.0
- Using previously-installed hashicorp/random v3.7.1

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Users\SCS\lab02>terraform plan
random_integer.rand: Refreshing state... [id=32318]
azurerm_resource_group.lab02_rg: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg]
azurerm_network_security_group.lab02_nsg1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_virtual_network.lab02_vnet: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# azurerm_container_group.lab02_container will be created
+ resource "azurerm_container_group" "lab02_container" {
  + dns_name_label           = "lab02-container-32318"
}
```

```
# azurerm_container_group.lab02_container will be created
+ resource "azurerm_container_group" "lab02_container" {
  + dns_name_label           = "lab02-container-32318"
  + dns_name_label_reuse_policy = "Unsecure"
  + exposed_port              = (known after apply)
  + fqdn                      = (known after apply)
  + id                        = (known after apply)
  + ip_address                = (known after apply)
  + ip_address_type           = "Public"
  + location                  = "eastus"
  + name                      = "lab02-container-32318"
  + network_profile_id        = (known after apply)
  + os_type                    = "Linux"
  + resource_group_name       = "lab02-rg"
  + restart_policy             = "Always"
  + sku                       = "Standard"
  + tags                      = {
      + "environment" = "lab"
    }

  + container {
    + commands = (known after apply)
    + cpu      = 1
    + image    = "mcr.microsoft.com/azuredocs/aci-helloworld"
    + memory   = 1.5
    + name     = "helloworld"

    + ports {
      + port    = 80
      + protocol = "TCP"
    }
  }
}

Plan: 1 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
```

```
C:\Users\SCS\lab02>terraform apply
random_integer.rand: Refreshing state... [id=32318]
azurerm_resource_group.lab02_rg: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg]
azurerm_virtual_network.lab02_vnet: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet]
azurerm_network_security_group.lab02_nsg1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# azurerm_container_group.lab02_container will be created
+ resource "azurerm_container_group" "lab02_container" {
    + dns_name_label           = "lab02-container-32318"
    + dns_name_label_reuse_policy = "Unsecure"
    + exposed_port              = (known after apply)
    + fqdn                      = (known after apply)
    + id                        = (known after apply)
    + ip_address                = "Public"
    + location                  = "eastus"
    + name                      = "Lab02-container-32318"
    + network_profile_id        = (known after apply)
    + os_type                   = "Linux"
    + resource_group_name       = "Lab02-rg"
    + restart_policy             = "Always"
    + sku                       = "Standard"
    + tags                      = {
        + "environment" = "lab"
    }

    + container {
        + commands = (known after apply)
        + cpu      = 1
    }
}

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
```

```
Enter a value: yes

azurerm_container_group.lab02_container: Creating...
azurerm_container_group.lab02_container: Still creating... [10s elapsed]
azurerm_container_group.lab02_container: Still creating... [20s elapsed]
azurerm_container_group.lab02_container: Still creating... [30s elapsed]
azurerm_container_group.lab02_container: Creation complete after 31s [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-container-32318]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

```
C:\Users\SCS\lab02>type terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.11.3",
  "serial": 12,
  "lineage": "826439df-592e-323e-e167-fe35fe767cee",
  "outputs": {},
  "resources": [
    {
      "mode": "managed",
      "type": "azurerm_container_group",
      "name": "lab02_container",
      "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "container": [
              {
                "commands": [],
                "cpu": 1,
                "cpu_limit": 0,
                "environment_variables": null,
                "image": "mcr.microsoft.com/azuredocs/aci-helloworld",
                "liveness_probe": [],
                "memory": 1.5,
                "memory_limit": 0,
                "name": "helloworld",
                "ports": [
                  {
                    "port": 80,
                    "protocol": "TCP"
                  }
                ],
                "readiness_probe": [],
                "secure_environment_variables": null,
                "security": [],
                "volume": []
              }
            ]
          }
        }
      ]
    }
  ]
}
```

```

      "security": [],
      "volume": []
    },
    "diagnostics": [],
    "dns_config": [],
    "dns_name_label": "lab02-container-32318",
    "dns_name_label_reuse_policy": "Unsecure",
    "exposed_port": [
      {
        "port": 80,
        "protocol": "TCP"
      }
    ],
    "fqdn": "lab02-container-32318.eastus.azurecontainer.io",
    "id": "subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-contains
ontainer-32318",
    "identity": [],
    "image_registry_credential": [],
    "init_container": [],
    "ip_address": "4.157.65.202",
    "ip_address_type": "Public",
    "key_vault_key_id": null,
    "key_vault_user_assigned_identity_id": null,
    "location": "eastus",
    "name": "lab02-container-32318",
    "network_profile_id": null,
    "os_type": "Linux",
    "priority": "",
    "resource_group_name": "lab02-rg",
    "restart_policy": "Always",
    "sku": "Standard",
    "subnet_ids": null,
    "tag": {
      "environment": "lab"
    },
    "timeouts": null,
    "zones": null
  ],
  "sensitive_attributes": [
  ]
}
```

```
"sensitive_attributes": [
  [
    {
      "type": "get_attr",
      "value": "container"
    },
    {
      "type": "index",
      "value": {
        "value": 0,
        "type": "number"
      }
    },
    {
      "type": "get_attr",
      "value": "secure_environment_variables"
    }
  ]
],
"private": "eyJlMmJyJczMC1lY2FhLTExZTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjozNjAwMDAwMDAwLCJkZWldGUiojE4MDAwMDAwMDAsInJlyWQiOjMwMDAwMDAwMCwidXBkYXRlIjozODAwMDAwMDAwfX0=",
"dependencies": [
  "azurerm_resource_group.lab02_rg",
  "random_integer.rand"
]
},
{
  "mode": "managed",
  "type": "azurerm_network_security_group",
  "name": "lab02_nsg1",
  "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
  "instances": [
    {
      "schema_version": 0,
      "attributes": {
        "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
      },
      "location": "eastus",
      "name": "lab02-nsg1",
    }
  ]
}
```

```
"location": "eastus",
"name": "lab02-nsg1",
"resource_group_name": "lab02-rg",
"security_rule": [
  {
    "access": "Allow",
    "description": "",
    "destination_address_prefix": "*",
    "destination_address_prefixes": [],
    "destination_application_security_group_ids": [],
    "destination_port_range": "22",
    "destination_port_ranges": [],
    "direction": "Inbound",
    "name": "SSH",
    "priority": 100,
    "protocol": "Tcp",
    "source_address_prefix": "*",
    "source_address_prefixes": [],
    "source_application_security_group_ids": [],
    "source_port_range": "*",
    "source_port_ranges": []
  }
],
"tags": {},
" timeouts": null
},
"sensitive_attributes": [],
"private": "eyJlMmJyJczMC1lY2FhLTExZTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjozODAwMDAwMDAwLCJkZWldGUiojE4MDAwMDAwMDAsInJlyWQiOjMwMDAwMDAwMCwidXBkYXRlIjozODAwMDAwMDAwfX0=",
"dependencies": [
  "azurerm_resource_group.lab02_rg"
]
},
{
  "mode": "managed",
  "type": "azurerm_resource_group",
  "name": "lab02_rg",
  "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
  "instances": [
    {
      "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg",
            "location": "eastus",
            "managed_by": "",
            "name": "lab02_rg",
            "tags": {},
            "timeouts": null
          },
          "sensitive_attributes": [],
          "private": "eyJlMmJyJczMC1lY2FhLTExZTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjo1NDAwMDAwMDAwLCJkZWldGUiojU0MDAwMDAwMDAsInJlyWQiOjMwMDAwMDAwMCwidXBkYXRlIjo1NDAwMDAwMDAwfX0="
        }
      ],
      "mode": "managed",
      "type": "azurerm_subnet",
      "name": "lab02_subnet1",
      "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "address_prefixes": [
              "10.0.1.0/24"
            ],
            "default_outbound_access_enabled": true,
            "delegation": [],
            "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnetS/lab02-subnet1",
            "name": "lab02-subnet1",
            "private_endpoint_network_policies": "Disabled",
            "private_link_service_network_policies_enabled": true,
            "resource_group_name": "lab02_rg",
            "service_endpoint_policy_ids": [],
            "service_endpoints": [],
            "timeouts": null,
            "tags": {}
          }
        }
      ]
    }
  ]
}
```

```
"provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
"instances": [
  {
    "schema_version": 0,
    "attributes": {
      "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg",
      "location": "eastus",
      "managed_by": "",
      "name": "lab02_rg",
      "tags": {},
      "timeouts": null
    },
    "sensitive_attributes": [],
    "private": "eyJlMmJyJczMC1lY2FhLTExZTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjo1NDAwMDAwMDAwLCJkZWldGUiojU0MDAwMDAwMDAsInJlyWQiOjMwMDAwMDAwMCwidXBkYXRlIjo1NDAwMDAwMDAwfX0="
  }
],
"mode": "managed",
" type": "azurerm_subnet",
" name": "lab02_subnet1",
" provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
" instances": [
  {
    "schema_version": 0,
    "attributes": {
      "address_prefixes": [
        "10.0.1.0/24"
      ],
      "default_outbound_access_enabled": true,
      "delegation": [],
      "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnetS/lab02-subnet1",
      "name": "lab02-subnet1",
      "private_endpoint_network_policies": "Disabled",
      "private_link_service_network_policies_enabled": true,
      "resource_group_name": "lab02_rg",
      "service_endpoint_policy_ids": [],
      "service_endpoints": [],
      "timeouts": null,
      "tags": {}
    }
  }
]
```

```

        "service_endpoints": [],
        "timeouts": null,
        "virtual_network_name": "lab02-vnet"
    },
    "sensitive_attributes": [],
    "private": "eyJlMmJyjczMC1lY2FhLTEzTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjoxODAwMDAwMDAwLCJkZWxldGUI0jE4MDAwMDAwMDAsInJLYWQjOjMwMDAwMDAwMCwidXBkYXRlIjoxODAwMDAwMDAwfX0=",
    "dependencies": [
        "azurerm_resource_group.lab02_rg",
        "azurerm_virtual_network.lab02_vnet"
    ]
},
{
    "mode": "managed",
    "type": "azurerm_subnet_network_security_group_association",
    "name": "lab02_nsg_assoc",
    "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
    "instances": [
        {
            "schema_version": 0,
            "attributes": {
                "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnet$lab02-subnet1",
                "network_security_group_id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1",
                "subnet_id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1",
                "timeouts": null
            },
            "sensitive_attributes": [],
            "private": "eyJlMmJyjczMC1lY2FhLTEzTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjoxODAwMDAwMDAwLCJkZWxldGUI0jE4MDAwMDAwMDAsInJLYWQjOjMwMDAwMDAwMH1",
            "dependencies": [
                "azurerm_network_security_group.lab02_nsg1",
                "azurerm_resource_group.lab02_rg",
                "azurerm_subnet.lab02_subnet1",
                "azurerm_virtual_network.lab02_vnet"
            ]
        }
    ]
}

```

```

        ],
    },
    "mode": "managed",
    "type": "azurerm_virtual_network",
    "name": "lab02_vnet",
    "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
    "instances": [
        {
            "schema_version": 0,
            "attributes": {
                "address_space": [
                    "10.0.0.0/16"
                ],
                "bgp_community": "",
                "ddos_protection_plan": [],
                "dns_servers": [],
                "edge_zone": "",
                "encryption": [],
                "flow_timeout_in_minutes": 0,
                "guid": "d95c9178-08f4-48fc-977d-0b6987dc7cf3",
                "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet",
                "location": "eastus",
                "name": "lab02-vnet",
                "private_endpoint_vnet_policies": "Disabled",
                "resource_group_name": "lab02-rg",
                "subnet": [
                    {
                        "address_prefixes": [
                            "10.0.1.0/24"
                        ],
                        "default_outbound_access_enabled": true,
                        "delegation": [],
                        "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1",
                        "name": "lab02-subnet1",
                        "private_endpoint_network_policies": "Disabled",
                        "private_link_service_network_policies_enabled": true,
                        "route_table_id": ""
                    }
                ]
            }
        }
    ]
}

```

```

        "service_endpoint_policy_ids": [],
        "service_endpoints": []
    },
    "tags": {},
    "timeouts": null
},
"sensitive_attributes": [],
"private": "eyJlMmJyjczMC1lY2FhLTEzTYtOGY4OC0zNDM2M2JjN2M0YzAiOnsiY3JlYXRlIjoxODAwMDAwMDAwLCJkZWxldGUI0jE4MDAwMDAwMDAsInJLYWQjOjMwMDAwMDAwMCwidXBkYXRlIjoxODAwMDAwMDAwfX0=",
"dependencies": [
    "azurerm_resource_group.lab02_rg"
],
},
{
    "mode": "managed",
    "type": "random_integer",
    "name": "rand",
    "provider": "provider[\"registry.terraform.io/hashicorp/random\"]",
    "instances": [
        {
            "schema_version": 0,
            "attributes": {
                "id": "32318",
                "keepers": null,
                "max": 99999,
                "min": 10000,
                "result": 32318,
                "seed": null
            },
            "sensitive_attributes": []
        }
    ],
    "check_results": null
}
C:\Users\SCS\lab02>

```

Part 4: Validate Configuration

8. Validate the configuration to ensure there are no errors or typos in the file:

```
terraform validate
```

9. Fix any issues in the lab02.tf file if reported (edit in your text editor).

10. Re-run the validation until no errors are reported.

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\Lab02>terraform validate
Success! The configuration is valid.
```

Part 5: Run Simulation

11. Perform a dry run:

```
terraform plan
```

12. Review output and ensure all configuration is as per requirements. Observe the resources with +, -, or -/+ signs.

13. Fix any issues in the lab02.tf file if reported (edit in your text editor).

14. Redo the dry run until no errors are reported:

```
terraform plan
```

Part 6: Deploy Infrastructure

15. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted to confirm.
-

Part 7: Get Information from Terraform State

16. View and analyze state information:

```
terraform state list  
terraform show
```

SCREENSHOT (capture both outputs).

```
C:\Users\SCS\lab02>terraform state list  
azurerm_container_group.lab02_container  
azurerm_network_security_group.lab02_nsg1  
azurerm_resource_group.lab02_rg  
azurerm_subnet.lab02_subnet1  
azurerm_subnet_network_security_group_association.lab02_nsg_assoc  
azurerm_virtual_network.lab02_vnet  
random_integer.rand  
  
C:\Users\SCS\lab02>terraform show  
# azurerm_container_group.lab02_container:  
resource "azurerm_container_group" "lab02_container" {  
  dns_name_label          = "lab02-container-32318"  
  dns_name_label_reuse_policy = "Unsecure"  
  exposed_port           = [  
    {  
      port     = 80  
      protocol = "TCP"  
    },  
  ]  
  fqdn                  = "lab02-container-32318.eastus.azurecontainer.io"  
  id                    = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/contain  
erGroups/lab02-container-32318"  
  ip_address            = "4.157.65.202"  
  ip_address_type       = "Public"  
  location              = "eastus"  
  name                  = "lab02-container-32318"  
  os_type                = "Linux"  
  priority              = null  
  resource_group_name   = "lab02-rg"  
  restart_policy         = "Always"  
  sku                   = "Standard"  
  tags                  = {  
    "environment" = "lab"  
  }  
  
  container {  
    commands     = []  
    cpu          = 1  
    cpu_limit    = 0  
    image        = "mcr.microsoft.com/azuredocs/aci-helloworld"  
  }  
  
  ports {  
    port          = 80  
    protocol     = "TCP"  
  }  
}  
  
# azurerm_network_security_group.lab02_nsg1:  
resource "azurerm_network_security_group" "lab02_nsg1" {  
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab  
02-nsg1"  
  location    = "eastus"  
  name        = "lab02-nsg1"  
  resource_group_name = "lab02-rg"  
  security_rule = [  
    {  
      access          = "Allow"  
      description    = null  
      destination_address_prefix = "*"  
      destination_address_prefixes = []  
      destination_application_security_group_ids = []  
      destination_port_range = "22"  
      destination_port_ranges = []  
      direction       = "Inbound"  
      name            = "SSH"  
      priority        = 100  
      protocol        = "Tcp"  
      source_address_prefix = "*"  
      source_address_prefixes = []  
      source_application_security_group_ids = []  
      source_port_range = "*"  
      source_port_ranges = []  
    },  
  ]  
  tags          = {}  
}
```

```
cpu_limit      = 0  
image          = "mcr.microsoft.com/azuredocs/aci-helloworld"  
memory         = 1.5  
memory_limit   = 0  
name           = "helloworld"  
  
ports {  
  port          = 80  
  protocol     = "TCP"  
}  
}  
  
# azurerm_network_security_group.lab02_nsg1:  
resource "azurerm_network_security_group" "lab02_nsg1" {  
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab  
02-nsg1"  
  location    = "eastus"  
  name        = "lab02-nsg1"  
  resource_group_name = "lab02-rg"  
  security_rule = [  
    {  
      access          = "Allow"  
      description    = null  
      destination_address_prefix = "*"  
      destination_address_prefixes = []  
      destination_application_security_group_ids = []  
      destination_port_range = "22"  
      destination_port_ranges = []  
      direction       = "Inbound"  
      name            = "SSH"  
      priority        = 100  
      protocol        = "Tcp"  
      source_address_prefix = "*"  
      source_address_prefixes = []  
      source_application_security_group_ids = []  
      source_port_range = "*"  
      source_port_ranges = []  
    },  
  ]  
  tags          = {}  
}
```

```

        ]
      tags          = {}

}

# azurerm_resource_group.lab02_rg:
resource "azurerm_resource_group" "lab02_rg" {
  id      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg"
  location  = "eastus"
  managed_by = null
  name      = "lab02-rg"
  tags      = {}
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
  address_prefixes           = [
    "10.0.1.0/24",
  ]
  default_outbound_access_enabled = true
  id                          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
  name                        = "lab02-subnet1"
  private_endpoint_network_policies = "Disabled"
  private_link_service_network_policies_enabled = true
  resource_group_name          = "lab02-rg"
  service_endpoint_policy_ids = []
  service_endpoints            = []
  virtual_network_name         = "lab02-vnet"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc" {
  id      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
  network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
  subnet_id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}

# azurerm_virtual_network.lab02_vnet:

```

```

# azurerm_virtual_network.lab02_vnet:
resource "azurerm_virtual_network" "lab02_vnet" {
  address_space           = [
    "10.0.0.0/16",
  ]
  bgp_community           = null
  dns_servers             = []
  edge_zone               = null
  flow_timeout_in_minutes = 0
  guid                    = "d95c9178-08f4-48fc-977d-0b6987dc7cf3"
  id                      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
  location                = "eastus"
  name                    = "lab02-vnet"
  private_endpoint_vnet_policies = "Disabled"
  resource_group_name     = "lab02-rg"
  subnet                 = [
    {
      address_prefixes           = [
        "10.0.1.0/24",
      ]
      default_outbound_access_enabled = true
      delegation                  = []
      id                          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
      name                        = "lab02-subnet1"
      private_endpoint_network_policies = "Disabled"
      private_link_service_network_policies_enabled = true
      route_table_id              = null
      security_group              = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
      service_endpoint_policy_ids = []
      service_endpoints            = []
    },
  ],
  tags                    = {}
}

# random_integer.rand:
resource "random_integer" "rand" {

```

```

# random_integer.rand:
resource "random_integer" "rand" {
  id      = "32318"
  max    = 99999
  min    = 10000
  result = 32318
}

```

C:\Users\SCS\lab02>

Part 8: Confirm Resource Creation in Azure

- Log in to the Azure Portal. Navigate to the resource group and confirm all resources exist as per the specifications.

SCREENSHOT (capture the Azure Portal showing the resource group and resources).

Name	Subscription	Location
lab02-rg	Azure for Students	East US
NetworkWatcherRG	Azure for Students	East US

Part 9: Expand the lab02.tf Script

20. Open lab02.tf in your text editor and add the following:

- Add another subnet to the virtual network called lab02-subnet2 using azurerm_subnet.
- Define a network security group called lab02-nsg2 with two inbound allow TCP rules:
 - Port 3389 called rule1 with priority 100.
 - Port 5985 called rule2 with priority 200 using azurerm_network_security_group.
- Attach the network security group lab02-nsg2 to lab02-subnet2 using azurerm_subnet_network_security_group_association.

SCREENSHOT of the Entire Terraform script file (capture the updated lab02.tf)

```
# Provider Block
provider "azurerm" {
  features {}

  subscription_id          = "ea6fc9c7-fe32-4565-b42a-e336c72040d9"
  resource_provider_registrations = "none"
}

# Random number to append to container name (optional but good practice)
resource "random_integer" "rand" {
  min = 10000
  max = 99999
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
  name      = "lab02-rg"
  location  = "East US"
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
  name          = "lab02-vnet"
  location      = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
  address_space    = ["10.0.0.0/16"]
}

# Subnet 1
resource "azurerm_subnet" "lab02_subnet1" {
  name          = "lab02-subnet1"
  resource_group_name = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes   = ["10.0.1.0/24"]
}

# Subnet 2 (New Subnet)
resource "azurerm_subnet" "lab02_subnet2" {
  name          = "lab02-subnet2"
  resource_group_name = azurerm_resource_group.lab02_rg.name
```

```

resource "azurerm_subnet" "lab02_subnet2" {
  name          = "lab02-subnet2"
  resource_group_name = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes     = ["10.0.2.0/24"] # New IP Range
}

# Network Security Group 1 (lab02-nsg1)
resource "azurerm_network_security_group" "lab02_nsg1" {
  name          = "lab02-nsg1"
  location      = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name

  security_rule {
    name          = "SSH"
    priority      = 100
    direction     = "Inbound"
    access        = "Allow"
    protocol      = "Tcp"
    source_port_range = "*"
    destination_port_range = "22"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }
}

# Network Security Group 2 (lab02-nsg2)
resource "azurerm_network_security_group" "lab02_nsg2" {
  name          = "lab02-nsg2"
  location      = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name

  security_rule {
    name          = "rule1"
    priority      = 100
    direction     = "Inbound"
    access        = "Allow"
    protocol      = "Tcp"
    source_port_range = "*"
    destination_port_range = "3389"
    source_address_prefix = "*"
  }
}

```

```

File Edit View
source_port_range      = "*"
destination_port_range = "3389"
source_address_prefix   = "*"
destination_address_prefix = "*"
}

security_rule {
  name          = "rule2"
  priority      = 200
  direction     = "Inbound"
  access        = "Allow"
  protocol      = "Tcp"
  source_port_range = "*"
  destination_port_range = "5985"
  source_address_prefix   = "*"
  destination_address_prefix = "*"
}
}

# Attach NSG1 to Subnet1
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
  subnet_id          = azurerm_subnet.lab02_subnet1.id
  network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}

# Attach NSG2 to Subnet2
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
  subnet_id          = azurerm_subnet.lab02_subnet2.id
  network_security_group_id = azurerm_network_security_group.lab02_nsg2.id
}

# Container Group (using Microsoft's public image)
resource "azurerm_container_group" "lab02_container" {
  name          = "lab02-container-${random_integer.rand.result}"
  location      = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
  os_type       = "Linux"
  ip_address_type = "Public"
  dns_name_label = "lab02-container-${random_integer.rand.result}"

  container {

```

```

container {
  name    = "helloworld"
  image   = "mcr.microsoft.com/azuredocs/aci-helloworld" #  UPDATED IMAGE
  cpu     = 1
  memory = 1.5
  ports {
    port    = 80
    protocol = "TCP"
  }
}

tags = {
  environment = "lab"
}
}
```

Part 10: Validate Configuration

21. Validate configuration to ensure there are no errors or typos:
22. Fix any issues in the lab02.tf file if reported.
23. Re-run the validation until no errors are reported:

SCREENTSHOT (capture the successful validation output).

```
C:\Users\SCS\lab02>terraform validate  
Success! The configuration is valid.
```

Part 11: Run Simulation

24. Perform a dry run:

```
terraform plan
```

25. Observe output closely. Note resources with +, -, or -/+ signs.

26. Fix any issues in the lab02.tf file if reported.

27. Redo the dry run until no errors are reported:

SCREENSHOT (capture the successful plan output).

```
C:\Users\SCS\lab02>terraform plan
random_integer.rand: Refreshing state... [id=32318]
azurerm_resource_group.lab02_rg: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_network_security_group.lab02_nsg1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_virtual_network.lab02_vnet: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet]
azurerm_container_group.lab02_container: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-container-32318]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
- destroy

Terraform will perform the following actions:

# azurerm_network_security_group.lab02_nsg2 will be created
+ resource "azurerm_network_security_group" "lab02_nsg2" {
  + id          = (known after apply)
  + location    = "eastus"
  + name        = "lab02-nsg2"
  + resource_group_name = "lab02-rg"
  + security_rule = [
    + {
      + access           = "Allow"
      + destination_address_prefix = "*"
      + destination_address_prefixes = []
      + destination_application_security_group_ids = []
      + destination_port_range     = "3389"
      + destination_port_ranges   = []
      + direction           = "Inbound"
      + name                = "rule1"
      + priority            = 100
      + protocol            = "Tcp"
      + source_address_prefix = "*"
    }
  ]
}
```

```

+ protocol                                = "Tcp"
+ source_address_prefix                  = "*"
+ source_address_prefixes                = []
+ source_application_security_group_ids = []
+ source_port_range                      = "*"
+ source_port_ranges                     = []
# (1 unchanged attribute hidden)
},
+
+ {
+ access                                 = "Allow"
+ destination_address_prefix           = "*"
+ destination_address_prefixes         = []
+ destination_application_security_group_ids = []
+ destination_port_range               = "5985"
+ destination_port_ranges              = []
+ direction                             = "Inbound"
+ name                                  = "rule2"
+ priority                             = 200
+ protocol                            = "Tcp"
+ source_address_prefix                 = "*"
+ source_address_prefixes               = []
+ source_application_security_group_ids = []
+ source_port_range                     = "*"
+ source_port_ranges                   = []
# (1 unchanged attribute hidden)
},
]
}
}

# azurerm_subnet.lab02_subnet2 will be created
+ resource "azurerm_subnet" "lab02_subnet2" {
+ address_prefixes                         = [
+   "10.0.2.0/24",
]
+ default_outbound_access_enabled          = true
+ id                                     = (known after apply)
+ name                                  = "lab02-subnet2"
+ private_endpoint_network_policies        = "Disabled"
+ private_link_service_network_policies_enabled = true
+ resource_group_name                     = "lab02-rg"
+ virtual_network_name                   = "lab02-vnet"

```

```

+ name                               = "lab02-subnet2"
+ private_endpoint_network_policies    = "Disabled"
+ private_link_service_network_policies_enabled = true
+ resource_group_name                 = "lab02-rg"
+ virtual_network_name                = "lab02-vnet"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc will be destroyed
# (because azurerm_subnet_network_security_group_association.lab02_nsg_assoc is not in configuration)
- resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc" {
-   id                                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks
/lab02-vnet/subnets/lab02-subnet1" -> null
-   network_security_group_id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurity
Groups/lab02-nsg1" -> null
-   subnet_id                          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks
/lab02-vnet/subnets/lab02-subnet1" -> null
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc1 will be created
+ resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
+   id                                = (known after apply)
+   network_security_group_id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurity
Groups/lab02-nsg1"
+   subnet_id                          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks
/lab02-vnet/subnets/lab02-subnet1"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc2 will be created
+ resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
+   id                                = (known after apply)
+   network_security_group_id          = (known after apply)
+   subnet_id                          = (known after apply)
}

```

plan: 4 to add, 0 to change, 1 to destroy.

Note: You didn't use the `-out` option to save this plan, so Terraform can't guarantee to take exactly these actions if you run `"terraform apply"` now.

C:\Users\SCS\lab02>

Part 12: Deploy Infrastructure

28. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted.

SCREENSHOT (scroll to the end if possible).

```
C:\Users\SCS\lab02>terraform apply
random_integer.rand: Refreshing state... [id=32318]
azurerm_resource_group.lab02_rg: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg]
azurerm_network_security_group.lab02_nsg1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_virtual_network.lab02_vnet: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet]
azurerm_network_security_group.lab02_nsg2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2]
azurerm_container_group.lab02_container: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-container-32318]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_subnet.lab02_subnet2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc1 will be created
+ resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
  + id = (known after apply)
  + network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
  + subnet_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes
```

```
Enter a value: yes
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Creating...
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Still creating... [10s elapsed]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Creation complete after 10s [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

C:\Users\SCS\lab02>
```

Part 13: Get Information from Terraform State

31. View and analyze state information:

```
terraform state list
```

SCREENSHOT (capture the output).

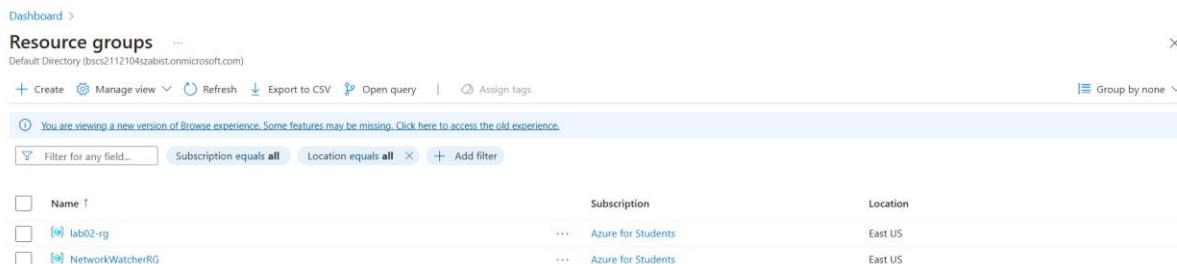
```
C:\Users\SCS\lab02>terraform state list
azurerm_container_group.lab02_container
azurerm_network_security_group.lab02_nsg1
azurerm_network_security_group.lab02_nsg2
azurerm_resource_group.lab02_rg
azurerm_subnet.lab02_subnet1
azurerm_subnet.lab02_subnet2
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2
azurerm_virtual_network.lab02_vnet
random_integer.rand
```

C:\Users\SCS\lab02>

Part 14: Confirm Resource Creation in Azure

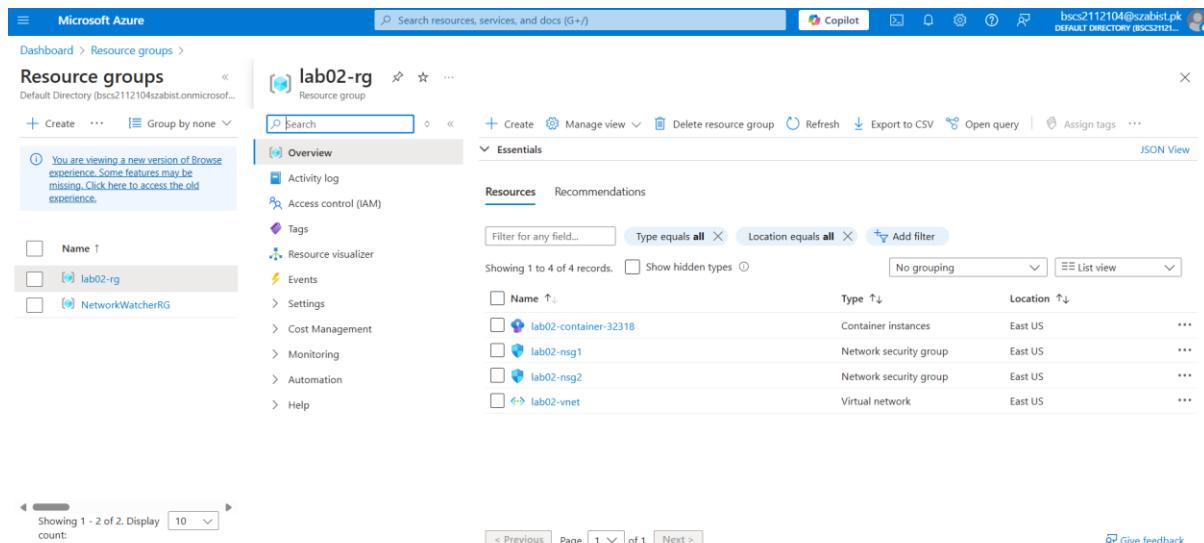
32. Log in to the Azure Portal. Navigate to the resource group and confirm all resources exist as per the specifications.

SCREENSHOT (capture the Azure Portal showing updated resources).



The screenshot shows the Azure Resource Groups page. It lists two resource groups: 'lab02-rg' and 'NetworkWatcherRG'. Both are located in 'East US' under 'Azure for Students' subscription. A message at the top indicates a new version of the browser is being used.

Name	Subscription	Location
lab02-rg	Azure for Students	East US
NetworkWatcherRG	Azure for Students	East US



The screenshot shows the 'lab02-rg' resource group overview. It lists four resources: 'lab02-container-32318' (Container instances), 'lab02-nsg1' (Network security group), 'lab02-nsg2' (Network security group), and 'lab02-vnet' (Virtual network). All resources are located in 'East US'.

Name	Type	Location
lab02-container-32318	Container instances	East US
lab02-nsg1	Network security group	East US
lab02-nsg2	Network security group	East US
lab02-vnet	Virtual network	East US

The screenshot shows the Microsoft Azure portal interface for managing a virtual network. The left sidebar navigation includes options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, Address space, Connected devices, and Subnets (which is currently selected). The main content area displays a table of subnets with columns for Name, IPv4, IPv6, Available IPs, Delegated to, Security group, and Route table. Two subnets are listed: lab02-subnet2 (IPv4 10.0.2.0/24) and lab02-subnet1 (IPv4 10.0.1.0/24). Both are delegated to 'lab02-nsg2' and have 251 available IPs. The URL in the browser bar is https://portal.azure.com/#@bscs2112104szabist.onmicrosoft.com/resource/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/overview.

The screenshot shows the Microsoft Azure portal interface for managing network security groups. The left sidebar navigation includes All services, Create, Manage view, Refresh, Export to CSV, Open query, and Assign tags. The main content area displays a table of network security groups (NSGs) with columns for Name, Resource group, Location, Subscription, and Flow log. Two NSGs are listed: lab02-nsg1 (Resource group: lab02-rg, Location: East US, Subscription: Azure for Students) and lab02-nsg2 (Resource group: lab02-rg, Location: East US, Subscription: Azure for Students). The URL in the browser bar is https://portal.azure.com/#@bscs2112104szabist.onmicrosoft.com/resource/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/list.

The screenshot shows the Microsoft Azure portal interface for managing container instances. The left sidebar navigation includes Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Settings, Containers, Identity, Properties, Locks, Monitoring, Automation, and Help. The main content area displays a table of container instances with columns for CPU (millicores) and Memory usage over time. One container instance is shown: lab02-container-32318 (Status: Running, Location: East US, Subscription: Azure for Students, IP address (Public): 4.157.65.202, FQDN: lab02-container-32318.eastus.azurecontainer.io, Container count: 1). The URL in the browser bar is https://portal.azure.com/#@bscs2112104szabist.onmicrosoft.com/resource/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerInstances/lab02-container-32318.

Part 15: Destroy All Resources and Verify

33. Destroy all the resources:

```
terraform destroy
```

- Type yes when prompted.

34. Verify deletion:

```
terraform state list
```

```
terraform show
```

SCREENSHOT (capture both outputs).

```
C:\Users\SCS\lab02>terraform destroy
random_integer.rand: Refreshing state... [id=32318]
azurerm_resource_group.lab02_rg: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg]
azurerm_network_security_group.lab02_nsg1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/networkSecurityGroups/lab02_nsg1]
azurerm_virtual_network.lab02_vnet: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/virtualNetworks/lab02_vnet]
azurerm_network_security_group.lab02_nsg2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/networkSecurityGroups/lab02_nsg2]
azurerm_container_group.lab02_container: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.ContainerInstance/containerGroups/lab02_container-32318]
azurerm_subnet.lab02_subnet2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/virtualNetworks/lab02_vnet/subnets/lab02_subnet2]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/virtualNetworks/lab02_vnet/subnets/lab02_subnet1]
azurerm_subnet.lab02_nsg2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/networkSecurityGroups/lab02_nsg2]
azurerm_subnet.network_security_group_association.lab02_nsg_assoc2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/virtualNetworks/lab02_vnet/subnets/lab02_subnet2]
azurerm_subnet.network_security_group_association.lab02_nsg_assoc1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/virtualNetworks/lab02_vnet/subnets/lab02_subnet1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
- destroy

Terraform will perform the following actions:

# azurerm_container_group.lab02_container will be destroyed
- resource "azurerm_container_group" "lab02_container" {
  - dns_name_label           = "lab02-container-32318" -> null
  - dns_name_label_reuse_policy = "Unsecure" -> null
  - exposed_port              = [
    - {
      - port      = 80
      - protocol = "TCP"
    },
  ] -> null
  - fqdn                    = "lab02-container-32318.eastus.azurecontainer.io" -> null
  - id                      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.ContainerInstance/containers/lab02-container-32318"
  - ip_address               = "4.157.65.202" -> null
  - ip_address_type          = "Public" -> null
  - location                 = "eastus" -> null
  - name                     = "lab02-container-32318" -> null
  - os_type                  = "Linux" -> null
  - resource_group_name       = "lab02_rg" -> null
  - restart_policy            = "Always" -> null
  - sku                      = "Standard" -> null
  - subnet_ids               = [] -> null
  - tags                     = {
    - "environment" = "lab"
  } -> null
  - zones                    = [] -> null
  # (1 unchanged attribute hidden)

  - container {
    - commands             = [] -> null
    - cpu                  = 1 -> null
    - cpu_limit             = 0 -> null
    - environment_variables = {} -> null
    - image                = "mcr.microsoft.com/azuredocs/aci-helloworld" -> null
    - memory               = 1.5 -> null
    - memory_limit          = 0 -> null
    - name                 = "helloworld" -> null
    - secure_environment_variables = (sensitive value) -> null
    - ports {
      - port      = 80 -> null
      - protocol = "TCP" -> null
    }
  }
}

# azurerm_network_security_group.lab02_nsg1 will be destroyed
- resource "azurerm_network_security_group" "lab02_nsg1" {
```



```
}

# azurerm_network_security_group.lab02_nsg1 will be destroyed
- resource "azurerm_network_security_group" "lab02_nsg1" {
  - id                      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/networkSecurityGroups/lab02_nsg1"
  - location                 = "eastus" -> null
  - name                     = "lab02-nsg1" -> null
  - resource_group_name       = "lab02_rg" -> null
  - security_rule             = [
    - {
      - access                = "Allow"
      - destination_address_prefix = "*"
      - destination_address_prefixes = []
      - destination_application_security_group_ids = []
      - destination_port_range     = "22"
      - destination_port_ranges   = []
      - direction               = "Inbound"
      - name                   = "SSH"
      - priority                = 100
      - protocol                = "TCP"
      - source_address_prefix   = "*"
      - source_address_prefixes = []
      - source_application_security_group_ids = []
      - source_port_range        = "*"
      - source_port_ranges       = []
    },
  ] -> null
  - tags                     = {} -> null
}

# azurerm_network_security_group.lab02_nsg2 will be destroyed
- resource "azurerm_network_security_group" "lab02_nsg2" {
  - id                      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02_rg/providers/Microsoft.Network/networkSecurityGroups/lab02_nsg2"
  - location                 = "eastus" -> null
  - name                     = "lab02-nsg2" -> null
  - resource_group_name       = "lab02_rg" -> null
  - security_rule             = [
```

```

Plan: 0 to add, 0 to change, 10 to destroy.

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]
azurerm_container_group.lab02_container: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-container-32318]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2: Destruction complete after 8s
azurerm_subnet.lab02_subnet2: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]
azurerm_network_security_group.lab02_nsg2: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2]
azurerm_container_group.lab02_container: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-.../containerGroups/lab02-container-32318, 10s elapsed]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet1, 10s elapsed]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Destruction complete after 14s
azurerm_subnet.lab02_subnet1: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_network_security_group.lab02_nsg1: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_network_security_group.lab02_nsg2: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...twork/networkSecurityGroups/lab02-nsg2, 10s elapsed]
azurerm_subnet.lab02_subnet2: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet2, 10s elapsed]
azurerm_container_group.lab02_container: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-.../containerGroups/lab02-container-32318, 20s elapsed]
azurerm_network_security_group.lab02_nsg2: Destruction complete after 13s
azurerm_container_group.lab02_container: Destruction complete after 22s

```

```

datanetworks,lab02_vnet,subnets,lab02_subnet2]
azurerm_network_security_group.lab02_nsg2: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2]
azurerm_container_group.lab02_container: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-.../containerGroups/lab02-container-32318, 10s elapsed]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet1, 10s elapsed]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Destruction complete after 14s
azurerm_subnet.lab02_subnet1: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_network_security_group.lab02_nsg1: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1]
azurerm_network_security_group.lab02_nsg2: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...twork/networkSecurityGroups/lab02-nsg2, 10s elapsed]
azurerm_subnet.lab02_subnet2: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet2, 10s elapsed]
azurerm_container_group.lab02_container: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-.../containerGroups/lab02-container-32318, 20s elapsed]
azurerm_network_security_group.lab02_nsg2: Destruction complete after 13s
azurerm_container_group.lab02_container: Destruction complete after 22s
random_integer.rand: Destroying... [id=32318]
random_integer.rand: Destruction complete after 0s
azurerm_network_security_group.lab02_nsg1: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...twork/networkSecurityGroups/lab02-nsg1, 10s elapsed]
azurerm_subnet.lab02_subnet1: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet1, 10s elapsed]
azurerm_subnet.lab02_subnet2: Destruction complete after 19s
azurerm_subnet.lab02_subnet1: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...works/lab02-vnet/subnets/lab02-subnet1, 20s elapsed]
azurerm_subnet.lab02_subnet1: Destruction complete after 25s
azurerm_virtual_network.lab02_vnet: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet]
azurerm_virtual_network.lab02_vnet: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-...oft.Network/virtualNetworks/lab02-vnet, 10s elapsed]
azurerm_virtual_network.lab02_vnet: Destruction complete after 13s
azurerm_resource_group.lab02_rg: Destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg]
azurerm_resource_group.lab02_rg: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg, 10s elapsed]
azurerm_resource_group.lab02_rg: Still destroying... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg, 20s elapsed]
azurerm_resource_group.lab02_rg: Destruction complete after 23s

Destroy complete! Resources: 18 destroyed.
C:\Users\SCS\lab02>

```

```

C:\Users\SCS\lab02>terraform state list

C:\Users\SCS\lab02>terraform show
The state file is empty. No resources are represented.

C:\Users\SCS\lab02>

```