Exercise 8– Creating a VPC in Terraform Objective:

Objective:

Learn how to use Terraform to create a basic Virtual Private Cloud (VPC) in AWS.
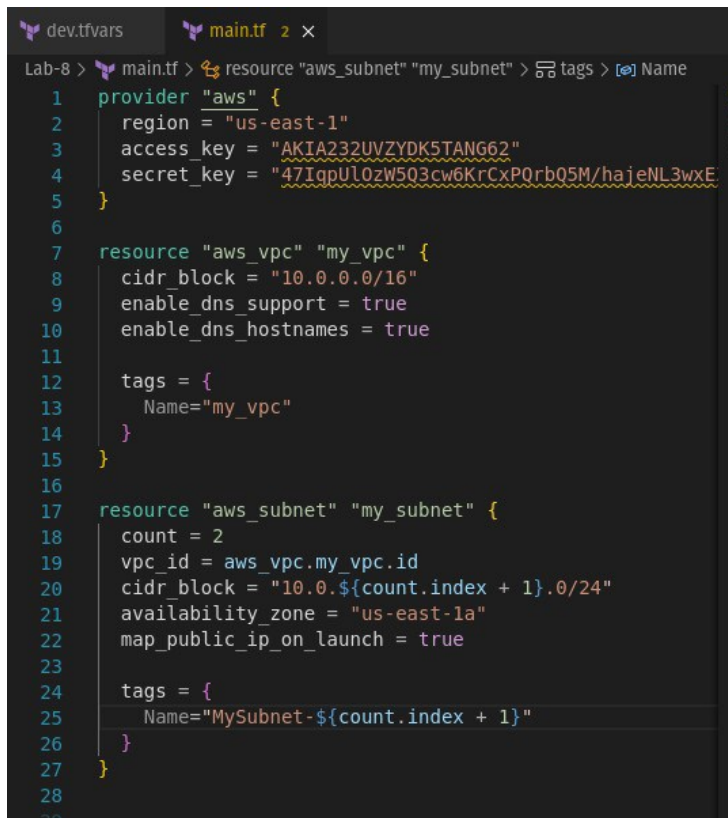
Prerequisites:

• Terraform installed on your machine.

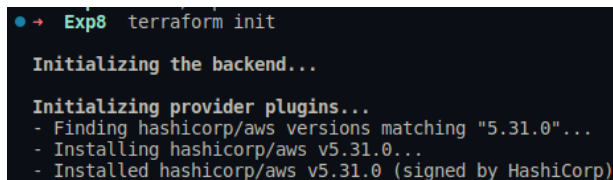• AWS CLI configured with the necessary credentials.

Steps:

1. Create a Terraform Directory:

• Create Terraform Configuration Files:

• Create a file named main.tf:

```
provider "aws" {
  region = "us-east-1"
  access_key = "AKIA232UVZYDK5TANG62"
  secret_key = "47IqpUlOzW5Q3cw6KrCxPQrbQ5M/hajeNL3wxE"
}

resource "aws_vpc" "my_vpc" {
  cidr_block = "10.0.0.0/16"
  enable_dns_support = true
  enable_dns_hostnames = true

  tags = {
    Name="my_vpc"
  }
}

resource "aws_subnet" "my_subnet" {
  count = 2
  vpc_id = aws_vpc.my_vpc.id
  cidr_block = "10.0.${count.index + 1}.0/24"
  availability_zone = "us-east-1a"
  map_public_ip_on_launch = true

  tags = {
    Name="MySubnet-${count.index + 1}"
  }
}
```

2. Initialize and Apply:

```
Exp8  terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (signed by HashiCorp)
```

```
Plan: 6 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

aws_vpc.ayroid: Creating...
aws_vpc.ayroid: Creation complete after 1s [id=vpc-09386b7a9c5369e7e]
aws_internet_gateway.ayroid-gw: Creating...
aws_subnet.ayroid-subnet: Creating...
aws_security_group.ayroid-sg: Creating...
aws_internet_gateway.ayroid-gw: Creation complete after 1s [id=igw-0b26c2f61998874d4]
aws_route_table.ayroid-rt: Creating...
aws_subnet.ayroid-subnet: Creation complete after 1s [id=subnet-09ba485855e0da2d4]
aws_route_table.ayroid-rt: Creation complete after 1s [id=rtb-06c0a2bbe2cbacda8]
aws_route_table_association.ayroid-rta: Creating...
aws_route_table_association.ayroid-rta: Creation complete after 0s [id=rtbassoc-09ab9a5e87f9be379]
aws_security_group.ayroid-sg: Creation complete after 2s [id=sg-0d55c5bc5f51ba0b9]

Apply complete! Resources: 6 added, 0 changed, 0 destroyed.
```

3. Verify Resources in AWS Console:

	Name	VPC ID	State	IPv4 CIDR	
☐	–	vpc-02c4e50a1abfffd3b	⊘ Available	172.31.0.0/16	–
☐	my_vpc	vpc-0c5f757b483b4d7ed	⊘ Available	10.0.0.0/16	–

	Name	Subnet ID	State
☐	MySubnet-2	subnet-0716cca6ad5e73a74	⊘ Available
☐	MySubnet-1	subnet-06be928c05048e8cf	⊘ Available

5. Clean Up:

```
aws_route_table_association.ayroid-rta: Destroying... [id=rtbassoc-06b27f63a6b7f489f]
aws_security_group.ayroid-sg: Destroying... [id=sg-0a13530fa8a50ee8e]
aws_route_table_association.ayroid-rta: Destruction complete after 1s
aws_subnet.ayroid-subnet: Destroying... [id=subnet-010c7aa4835e8b4f0]
aws_route_table.ayroid-rt: Destroying... [id=rtb-0c7a93722671efd9f]
aws_security_group.ayroid-sg: Destruction complete after 1s
aws_subnet.ayroid-subnet: Destruction complete after 0s
aws_route_table.ayroid-rt: Destruction complete after 0s
aws_internet_gateway.ayroid-gw: Destroying... [id=igw-0ee28d334c38b7999]
aws_internet_gateway.ayroid-gw: Destruction complete after 1s
aws_vpc.ayroid: Destroying... [id=vpc-06a6d2f68c04e7ecf]
aws_vpc.ayroid: Destruction complete after 0s

Destroy complete! Resources: 6 destroyed.
```