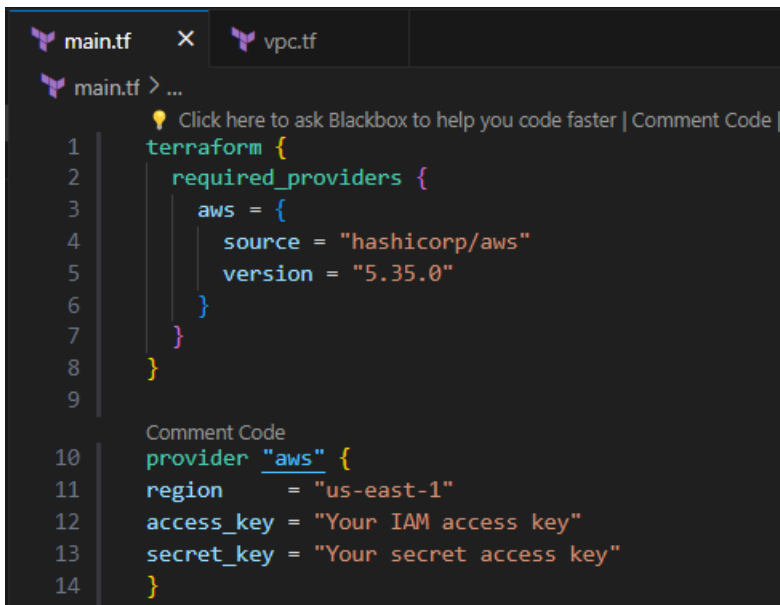


## EXPERIMENT – 8

Name: - Shashwat. Dnyaneshwar Kamdi
Batch – 2 [DevOps Non-Hons]
SAP ID- 500092140
Subject – System Provisioning and Configuration Management Lab

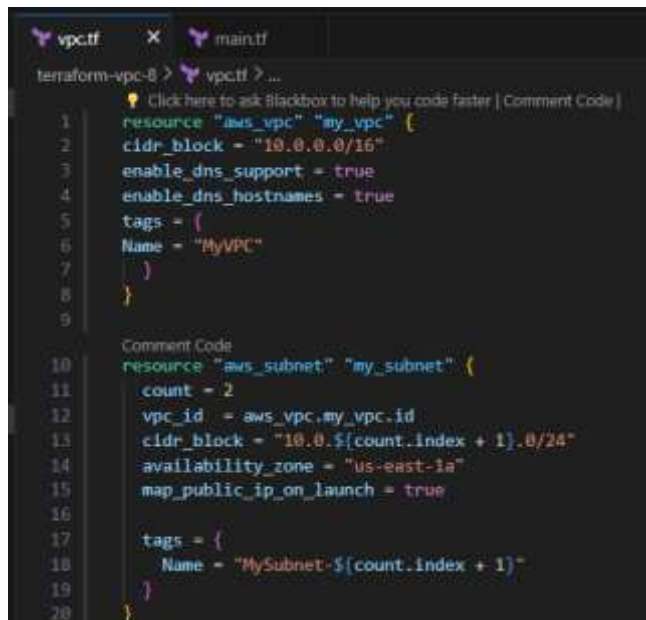
### **Aim: Creating a VPC in Terraform**

1] Create a Terraform Configuration File (main.tf)



```
main.tf > ...
1 terraform {
2     required_providers {
3         aws = {
4             source = "hashicorp/aws"
5             version = "5.35.0"
6         }
7     }
8 }
9
10 provider "aws" {
11     region     = "us-east-1"
12     access_key = "Your IAM access key"
13     secret_key = "Your secret access key"
14 }
```

2] Create new file name as “vpc.tf”



```
vpc.tf > ...
1 resource "aws_vpc" "my_vpc" {
2     cidr_block = "10.0.0.0/16"
3     enable_dns_support = true
4     enable_dns_hostnames = true
5     tags = {
6         Name = "MyVPC"
7     }
8 }
9
10 resource "aws_subnet" "my_subnet" {
11     count = 2
12     vpc_id = aws_vpc.my_vpc.id
13     cidr_block = "10.0.${count.index + 1}.0/24"
14     availability_zone = "us-east-1a"
15     map_public_ip_on_launch = true
16
17     tags = {
18         Name = "MySubnet-${count.index + 1}"
19     }
20 }
```

### 3] Initialize Terraform using command “terraform init”

```
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\terraform-vpc-8> terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

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.
```

### 4] Apply it using command “Terraform apply -auto-approve”

```
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\terraform-vpc-8> terraform apply -auto-approve
aws_vpc.my_vpc: Refreshing state... [id=vpc-04e8239d5af67a047]

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:

# aws_subnet.my_subnet[0] will be created
+ resource "aws_subnet" "my_subnet" {
  + arn                                     = (known after apply)
  + assign_ipv6_address_on_creation        = false
  + availability_zone                      = "us-east-1a"
  + availability_zone_id                   = (known after apply)
  + cidr_block                             = "10.0.1.0/24"
  + enable_dns64                           = false
  + enable_resource_name_dns_a_record_on_launch = false
  + enable_resource_name_dns_aaaa_record_on_launch = false
  + id                                     = (known after apply)
  + ipv6_cidr_block_association_id         = (known after apply)
  + ipv6_native                             = false
  + map_public_ip_on_launch                = true
  + owner_id                               = (known after apply)
  + private_dns_hostname_type_on_launch    = (known after apply)
  + tags                                   = {
    + "Name" = "MySubnet-1"
  }
  + tags_all                               = {
    + "Name" = "MySubnet-1"
  }
  + vpc_id                                 = "vpc-04e8239d5af67a047"
}

# aws_subnet.my_subnet[1] will be created
+ resource "aws_subnet" "my_subnet" {
  + arn                                     = (known after apply)
  + assign_ipv6_address_on_creation        = false
  + availability_zone                      = "us-east-1a"
  + availability_zone_id                   = (known after apply)
  + cidr_block                             = "10.0.2.0/24"
  + enable_dns64                           = false
  + enable_resource_name_dns_a_record_on_launch = false
  + enable_resource_name_dns_aaaa_record_on_launch = false
  + id                                     = (known after apply)
  + ipv6_cidr_block_association_id         = (known after apply)
  + ipv6_native                             = false
  + map_public_ip_on_launch                = true
  + owner_id                               = (known after apply)
  + private_dns_hostname_type_on_launch    = (known after apply)
  + tags                                   = {
    + "Name" = "MySubnet-2"
  }
  + tags_all                               = {
    + "Name" = "MySubnet-2"
  }
  + vpc_id                                 = "vpc-04e8239d5af67a047"
}

Plan: 2 to add, 0 to change, 0 to destroy.
aws_subnet.my_subnet[0]: Creating...
aws_subnet.my_subnet[1]: Creating...
aws_subnet.my_subnet[1]: Still creating... [10s elapsed]
aws_subnet.my_subnet[0]: Still creating... [10s elapsed]
aws_subnet.my_subnet[1]: Creation complete after 14s [id=subnet-07f4068fdbbd753a5]
aws_subnet.my_subnet[0]: Creation complete after 14s [id=subnet-0c4465c01bb374d9d]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\terraform-vpc-8>
```

## 5] Verify Resources on AWS Management Console.

**Your VPCs (2)** [Info](#)

Search

<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR
<input type="checkbox"/>	MyVPC	<a href="#">vpc-04e8239d5af67a047</a>	Available	10.0.0.0/16
<input type="checkbox"/>	-	<a href="#">vpc-0360c70e9cdff24526</a>	Available	172.31.0.0/16

VPC > Your VPCs > [vpc-04e8239d5af67a047](#)

**vpc-04e8239d5af67a047 / MyVPC** [Actions](#)

**Details** [Info](#)

VPC ID	<a href="#">vpc-04e8239d5af67a047</a>	State	Available	DNS hostnames	Enabled	DNS resolution	Enabled
Tenancy	Default	DHCP options set	<a href="#">dopt-0c4e6c5e2617e94b</a>	Main route table	<a href="#">rtb-0f852d57d074e8711</a>	Main network ACL	<a href="#">acl-0275e51fc08d330f2</a>
Default VPC	No	IPv4 CIDR	10.0.0.0/16	IPv6 pool	-	IPv6 CIDR (Network border group)	-
Network Address Usage metrics	Enabled	Route 53 Resolver DNS Firewall rule	-	Owner ID	<a href="#">394464380823</a>		

## 6] Cleanup Resources using command “Terraform destroy”

```
# aws_vpc.my_vpc will be destroyed
resource "aws_vpc" "my_vpc" {
  arn = "arn:aws:ec2:us-east-1:394464380823:vpc/vpc-04e8239d5af67a047" -> null
  assign_generated_ipv6_cidr_block = false -> null
  cidr_block = "10.0.0.0/16" -> null
  default_network_acl_id = "acl-0275e51fc08d330f2" -> null
  default_route_table_id = "rtb-0f852d57d074e8711" -> null
  default_security_group_id = "sg-0113ff9d60c4dcb59" -> null
  dhcp_options_id = "dopt-0c4e6c5e2617e94b" -> null
  enable_dns_hostnames = true -> null
  enable_dns_support = true -> null
  enable_network_address_usage_metrics = false -> null
  id = "vpc-04e8239d5af67a047" -> null
  instance_tenancy = "default" -> null
  ipv6_netmask_length = 0 -> null
  main_route_table_id = "rtb-0f852d57d074e8711" -> null
  owner_id = "394464380823" -> null
  tags = {
    Name = "MyVPC"
  } -> null
  tags_all = {
    Name = "MyVPC"
  } -> null
}

Plan: 0 to add, 0 to change, 3 to destroy.
aws_subnet.my_subnet[1]: Destroying... [id=subnet-07f4868fdbbd753a5]
aws_subnet.my_subnet[0]: Destroying... [id=subnet-0c4485c81bb374d9d]
aws_subnet.my_subnet[1]: Destruction complete after 2s
aws_subnet.my_subnet[0]: Destruction complete after 2s
aws_vpc.my_vpc: Destroying... [id=vpc-04e8239d5af67a047]
aws_vpc.my_vpc: Destruction complete after 3s

Destroy complete! Resources: 3 destroyed.
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\terraform-vpc-B>
```

## 7] Now Update the vpc.tf file and repeat the previous Steps.

```
resource "aws_vpc" "gfg-vpc" {
  cidr_block = "10.0.0.0/16"
}

Comment Code
resource "aws_subnet" "gfg-subnet" {
  vpc_id = aws_vpc.gfg-vpc.id
  cidr_block = "10.0.1.0/24"

  tags = {
    Name = "gfg-subnet"
  }
}

Comment Code
resource "aws_internet_gateway" "gfg-gw" {
  vpc_id = aws_vpc.gfg-vpc.id

  tags = {
    Name = "gfg-IG"
  }
}
```

```

Comment Code
resource "aws_route_table" "gfg-rt" {
  vpc_id = aws_vpc.gfg-vpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.gfg-gw.id
  }

  tags = {
    Name = "GFG-Route-Table"
  }
}

```

```

Comment Code
resource "aws_route_table_association" "gfg-rta" {
  subnet_id      = aws_subnet.gfg-subnet.id
  route_table_id = aws_route_table.gfg-rt.id
}

```

```

Comment Code
resource "aws_security_group" "gfg-sg" {
  name      = "my-gfg-sg"
  vpc_id    = aws_vpc.gfg-vpc.id
}

```

```

ingress {
  description      = "TLS from VPC"
  from_port        = 20
  to_port          = 20
  protocol         = "tcp"
  cidr_blocks      = ["0.0.0.0/0"]
  ipv6_cidr_blocks = [ ":::/0" ]
}

egress {
  from_port        = 0
  to_port          = 0
  protocol         = "-1"
  cidr_blocks      = ["0.0.0.0/0"]
  ipv6_cidr_blocks = [ ":::/0" ]
}

tags = {
  Name = "my-gfg-sg"
}
}

```

8] Apply it using command “Terraform apply”

```

# aws_vpc.gfg-vpc will be created
+ resource "aws_vpc" "gfg-vpc" {
+   cidr_block                  = (known after apply)
+   default_network_acl_id     = (known after apply)
+   default_route_table_id     = (known after apply)
+   default_security_group_id  = (known after apply)
+   dhcp_options_id            = (known after apply)
+   enable_dns_hostnames        = (known after apply)
+   enable_dns_support          = true
+   enable_network_address_usage_metrics = (known after apply)
+   id                          = (known after apply)
+   instance_tenancy            = "default"
+   ipv6_association_id         = (known after apply)
+   ipv6_cidr_block             = (known after apply)
+   ipv6_cidr_block_network_border_group = (known after apply)
+   main_route_table_id         = (known after apply)
+   nat_gateway_id              = (known after apply)
+   tags_all                    = (known after apply)
+ }

Plan: 0 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.gfg-vpc: Creating...
aws_vpc.gfg-vpc: Creation complete after 1s [id=vpc-037ceb0d28d510d30]
aws_internet_gateway.gfg-gw: Creating...
aws_subnet.gfg-subnet: Creating...
aws_security_group.gfg-sg: Creating...
aws_internet_gateway.gfg-gw: Creation complete after 0s [id=igw-0017c30c9e5f43f2c]
aws_route_table.gfg-rt: Creating...
aws_subnet.gfg-subnet: Creation complete after 0s [id=subnet-0f4ca76fc730dc0]
aws_route_table.gfg-rt: Creation complete after 1s [id=rth-0320a00a0c1cec2f]
aws_route_table_association.gfg-rta: Creating...
aws_route_table_association.gfg-rta: Creation complete after 0s [id=rthassoc-079ea0900bc41e0ee]
aws_security_group.gfg-sg: Creation complete after 2s [id=sg-00923d6d75a1674b5]

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

F:\New 6\SPCM_LAB\spcm-lab-terraform\src_03

```

9] Again verify resources on AWS Management Console.

Find resources by attribute or tag					
<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR
<input type="checkbox"/>	-	<a href="#">subnet-01570740224111d49</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.64.0/20
<input type="checkbox"/>	-	<a href="#">subnet-0250a101a3bf4d611</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.80.0/20
<input type="checkbox"/>	-	<a href="#">subnet-00ffbe2eff0b4f519</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.32.0/20
<input type="checkbox"/>	-	<a href="#">subnet-03bc5a8ea2c1d1e42</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.48.0/20
<input type="checkbox"/>	-	<a href="#">subnet-0b91288e314d7187a</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.0.0/20
<input type="checkbox"/>	-	<a href="#">subnet-02503ea72be5c83b6</a>	Available	<a href="#">vpc-0160c70ecbdf24526</a>	172.31.16.0/20
<input type="checkbox"/>	gfg-subnet	<a href="#">subnet-0756225df2d5bc2b4</a>	Available	<a href="#">vpc-0197e1225b97f79e0</a>	10.0.1.0/24

## 10] Cleanup Resources using command “Terraform destroy”

```

- tags_all
}

= {} -> null

Plan: 0 to add, 0 to change, 6 to destroy.
aws_route_table_association.gfg-rt: Destroying... [id=rtbassoc-0d95621e1ea5b363a]
aws_security_group.gfg-sg: Destroying... [id=sg-098e9fc67acef8608]
aws_route_table_association.gfg-rt: Destruction complete after 2s
aws_subnet.gfg-subnet: Destroying... [id=subnet-0769225df2d5be2b4]
aws_route_table.gfg-rt: Destroying... [id=rtb-0ca5f00981cd488ce]
aws_security_group.gfg-sg: Destruction complete after 2s
aws_subnet.gfg-subnet: Destruction complete after 1s
aws_route_table.gfg-rt: Destruction complete after 2s
aws_internet_gateway.gfg-gw: Destroying... [id=igw-01a93c930fa4df43b]
aws_internet_gateway.gfg-gw: Destruction complete after 1s
aws_vpc.gfg-vpc: Destroying... [id=vpc-0397e1225b97f79e0]
aws_vpc.gfg-vpc: Destruction complete after 2s

Destroy complete! Resources: 6 destroyed.
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Exp -8>

```