Experiment No. 2.3

Student Name: Rishav Kumar   **UID: 22MCC20039**

Branch: MCA - CCD   **Section/Group: MCD-1/ Grp A**

Semester: IV   **Date of Performance: 07ᵗʰ Mar 24**

Subject Name: CI/CD Pipelines   **Subject Code: 22CAP-781**

1. Aim/Overview of the practical:

a. Create AWS free tier account

b. Install Terraform in local system

c. Initialize terraform and create VPC, Public Subnet, Internet Gateways, Routing Table and create 3 instances inside VPC.

d. Execute terraform plan and apply the changes using terraform apply.

e. After performing step c., destroy all the instances.

2. Code for practical:

a. Create AWS free tier account:

- Go to the AWS Free Tier website: AWS Free Tier.
- Click on the "Create an AWS Account" button and follow the instructions to set up AWS account.

b. Install Terraform in the local system:

- Download the Terraform binary for your operating system from the official Terraform website.
- Extract the downloaded archive to a directory.
- Add the directory containing the Terraform binary to your system's PATH.

c. Initialize Terraform and create 3 instances on AWS:

- Create a new directory for your Terraform configuration.
- Inside the directory, create a file named **main.tf** with the following content

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.0"
    }
```
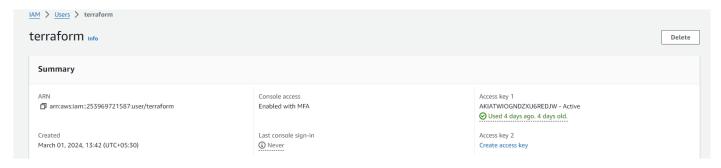
```
  }
}

resource "aws_instance" "web" {
  ami           = "ami-00381a880aa48c6c6"
  instance_type = "t3.micro"

  tags = {
    Name = "Arkay"
  }
}

provider "aws" {
  region     = "eu-north-1"
  access_key = "AKIATWIOGNDZXU6REDJW"
  secret_key = "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"
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/24"
}

resource "aws_subnet" "main" {
  vpc_id     = aws_vpc.main.id
  cidr_block = "10.0.0.0/24"

  tags = {
    Name = "Arkay"
  }
}

resource "aws_internet_gateway" "gw" {
  vpc_id = aws_vpc.main.id

  tags = {
    Name = "Arkay"
  }
}

resource "aws_route_table" "bar" {
  vpc_id = aws_vpc.main.id

  route {
    cidr_block = "10.0.0.0/16"
    gateway_id = aws_internet_gateway.gw.id
  }

  tags = {
    Name = "Arkay"
  }
}


resource "aws_route_table_association" "a" {
  subnet_id      = aws_subnet.main.id
  route_table_id = aws_route_table.bar.id
}
```

- Go to IAM, and create a user and create access key and secret key.

IAM > Users > terraform

terraform Info

[Delete]

Summary

ARN	Console access	Access key 1
🗐 arn:aws:iam::253969721587:user/terraform	Enabled with MFA	AKIATWIOGNDZXU6REDJW - Active
		⊘ Used 4 days ago. 4 days old.
Created	Last console sign-in	Access key 2
March 01, 2024, 13:42 (UTC+05:30)	⊕ Never	Create access key

- Open a terminal, navigate to the directory, and run the following command:

terraform init

d. Execute Terraform plan and apply the changes:

terraform plan

```
# aws_vpc.main will be created
+ resource "aws_vpc" "main" {
    + arn                                  = (known after apply)
    + cidr_block                           = "10.0.0.0/24"
    + 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)
    + owner_id                             = (known after apply)
    + tags_all                             = (known after apply)
  }

Plan: 6 to add, 0 to change, 0 to destroy.
```
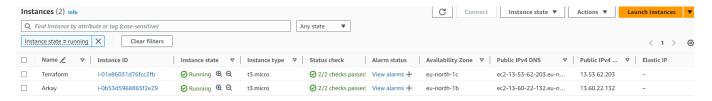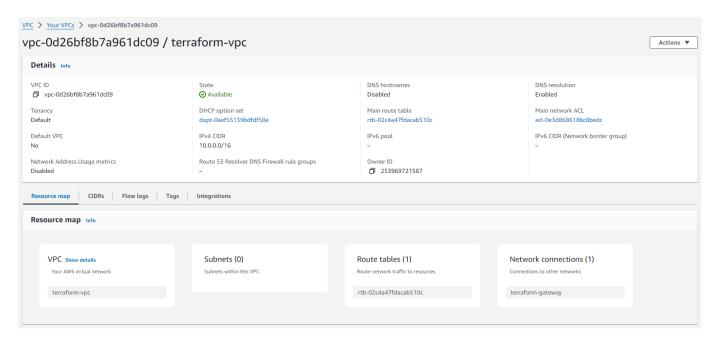
terraform apply

```
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.main: Creating...
aws_instance.web: Creating...
aws_vpc.main: Creation complete after 2s [id=vpc-08350de48c6b78d57]
aws_internet_gateway.gw: Creating...
aws_subnet.main: Creating...
aws_internet_gateway.gw: Creation complete after 0s [id=igw-08c6c3200749fb7c1]
aws_route_table.bar: Creating...
aws_subnet.main: Creation complete after 0s [id=subnet-0cd495a212047d411]
aws_route_table.bar: Creation complete after 1s [id=rtb-01920129b3ee6052f]
aws_route_table_association.a: Creating...
aws_route_table_association.a: Creation complete after 0s [id=rtbassoc-0b92b25273e1e47be]
aws_instance.web: Still creating... [10s elapsed]
aws_instance.web: Creation complete after 13s [id=i-03fc2370c0e1575f1]

Apply complete! Resources: 6 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-21-76:~/terraform$
```

Created Instance

	Name ▲	▽	Instance ID	Instance state	▽	Instance type	▽	Status check	Alarm status	Availability Zone	▽	Public IPv4 DNS	▽	Public IPv4 ...	▽	Elastic IP
☐	Terraform		i-01e86031d76fcc2fb	⊘ Running ⊕ ⊖		t3.micro		⊘ 2/2 checks passed	View alarms ╋	eu-north-1c		ec2-13-53-62-203.eu-n...		13.53.62.203		–
☐	Arkay		i-0b53d5968865f2e29	⊘ Running ⊕ ⊖		t3.micro		⊘ 2/2 checks passed	View alarms ╋	eu-north-1b		ec2-13-60-22-132.eu-n...		13.60.22.132		–

Instances (2) Info — Find Instance by attribute or tag (case-sensitive) — Any state — Instance state = running ✕ — Clear filters — ⟳ Connect — Instance state ▼ — Actions ▼ — **Launch instances** ▼ — ‹ 1 › ⚙

Created VPC

VPC › Your VPCs › vpc-0d26bf8b7a961dc09

vpc-0d26bf8b7a961dc09 / terraform-vpc   Actions ▼

Details Info

VPC ID	State	DNS hostnames	DNS resolution
⎘ vpc-0d26bf8b7a961dc09	⊘ Available	Disabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-0aef55139bdfdf58e	rtb-02c4a47fdacab510c	acl-0e3d868618bc8bedc
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	10.0.0.0/16	–	–
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	
Disabled	–	⎘ 253969721587	

Resource map | CIDRs | Flow logs | Tags | Integrations

Resource map Info

VPC Show details	Subnets (0)	Route tables (1)	Network connections (1)
Your AWS virtual network	Subnets within this VPC	Route network traffic to resources	Connections to other networks
terraform-vpc		rtb-02c4a47fdacab510c	terraform-gateway

Created Gateway

igw-0e76b3263cbaf1cae / terraform-gateway

Actions ▼

Details Info

Internet gateway ID	State	VPC ID	Owner
📋 igw-0e76b3263cbaf1cae	✓ Attached	vpc-0d26bf8b7a961dc09	terraform-vpc	📋 253969721587

Tags

Manage tags

🔍 Search tags

< 1 > ⚙

Key	Value
Name	terraform-gateway

e. Destroy all the instances:

terraform destroy

```
Plan: 0 to add, 0 to change, 6 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

aws_route_table_association.a: Destroying... [id=rtbassoc-0b92b25273e1e47be]
aws_instance.web: Destroying... [id=i-03fc2370c0e1575f1]
aws_route_table_association.a: Destruction complete after 0s
aws_route_table.bar: Destroying... [id=rtb-01920129b3ee6052f]
aws_subnet.main: Destroying... [id=subnet-0cd495a212047d411]
aws_subnet.main: Destruction complete after 0s
aws_route_table.bar: Destruction complete after 0s
aws_internet_gateway.gw: Destroying... [id=igw-08c6c3200749fb7c1]
aws_internet_gateway.gw: Destruction complete after 1s
aws_vpc.main: Destroying... [id=vpc-08350de48c6b78d57]
aws_vpc.main: Destruction complete after 0s
aws_instance.web: Still destroying... [id=i-03fc2370c0e1575f1, 10s elapsed]
aws_instance.web: Still destroying... [id=i-03fc2370c0e1575f1, 20s elapsed]
aws_instance.web: Still destroying... [id=i-03fc2370c0e1575f1, 30s elapsed]
aws_instance.web: Destruction complete after 30s

Destroy complete! Resources: 6 destroyed.
ubuntu@ip-172-31-21-76:~/terraform$
```