Name – Majahar Mahamud Kazi

Div – B **Batch – B2**

Roll no. – 322036 **PRN no. – 22110729**

Assignment 5

Aim: Write IaC using terraform to create EC2 machine on aws or azure or google cloud.

Theory:

A. What is Terraform ?

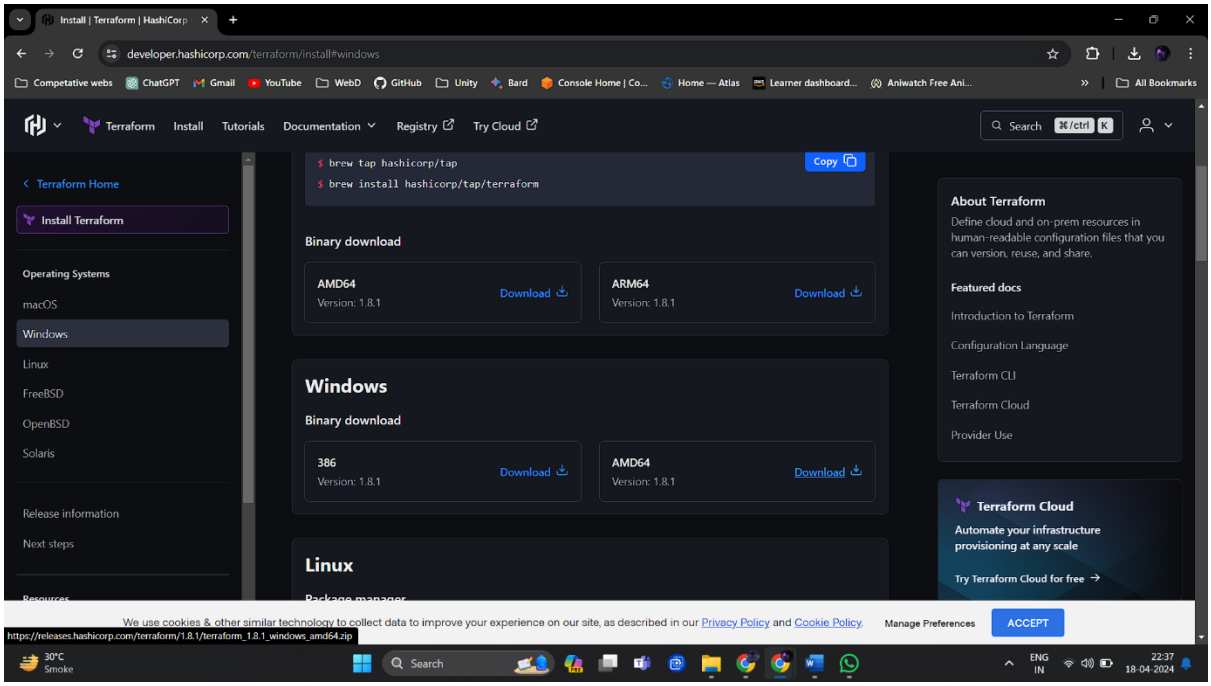
Terraform is an open-source infrastructure as code (laC) tool developed by HashiCorp. It allows developers to define and manage their infrastructure in a declarative way, using configuration files instead of manual processes.

With Terraform, infrastructure changes can be easily versioned, tested, and deployed across multiple cloud providers, such as AWS, Google Cloud Platform, and Microsoft Azure. Terraform's modular design allows for easy reuse of infrastructure code and promotes collaboration among team members. It automates the deployment of infrastructure, making it more efficient and reliable, reducing the risk of human error.

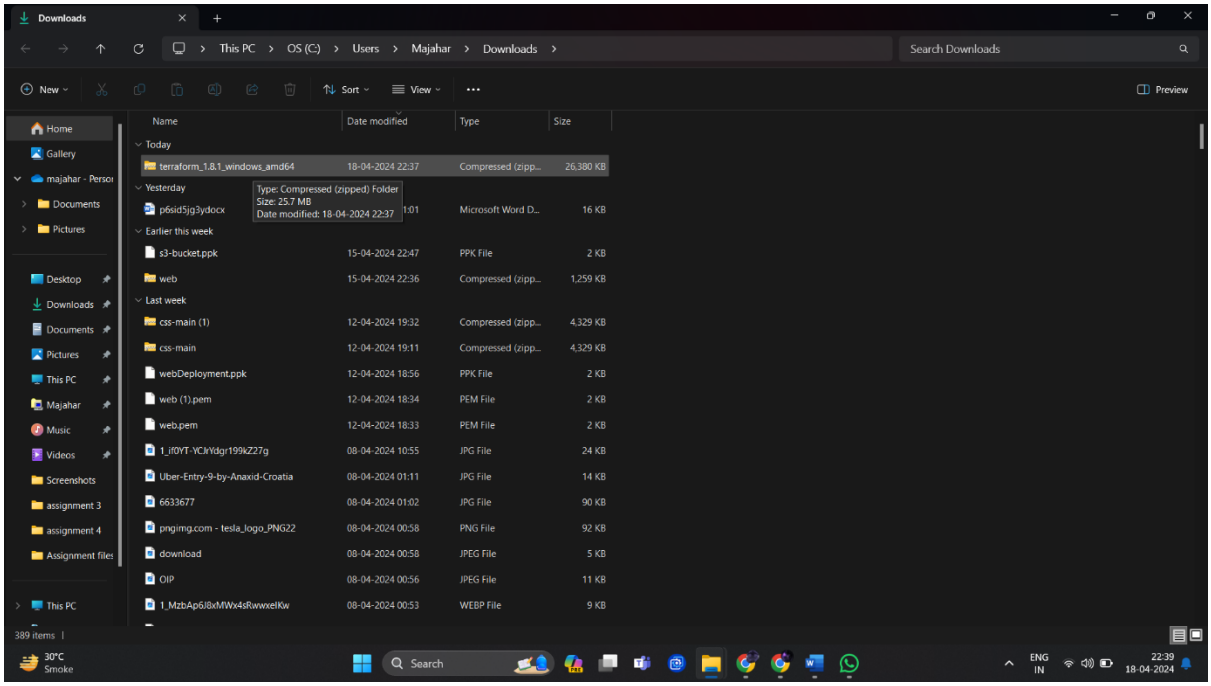
Terraform helps teams to manage their infrastructure as code, and provides greater agility and scalability to their projects.

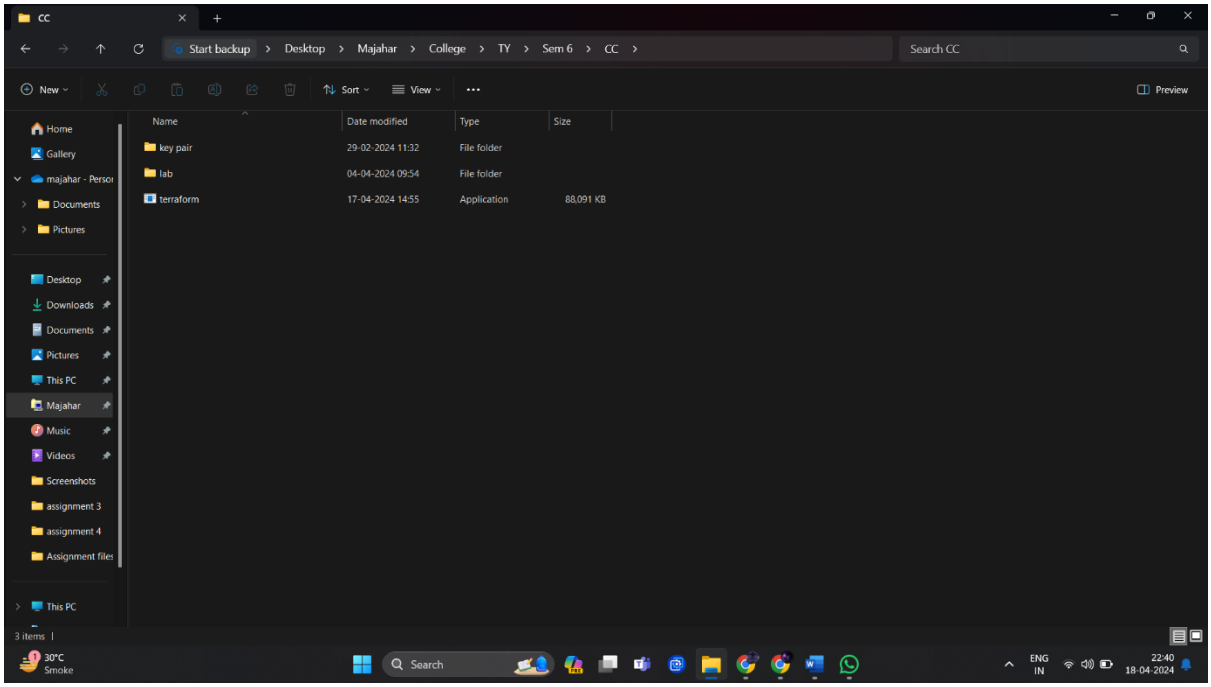
B. Step-by-step screenshot to install and configure Terraform:

To download Terraform first go to link - https://developer.hashicorp.com/terraform/downloads and select required terraform version to download.

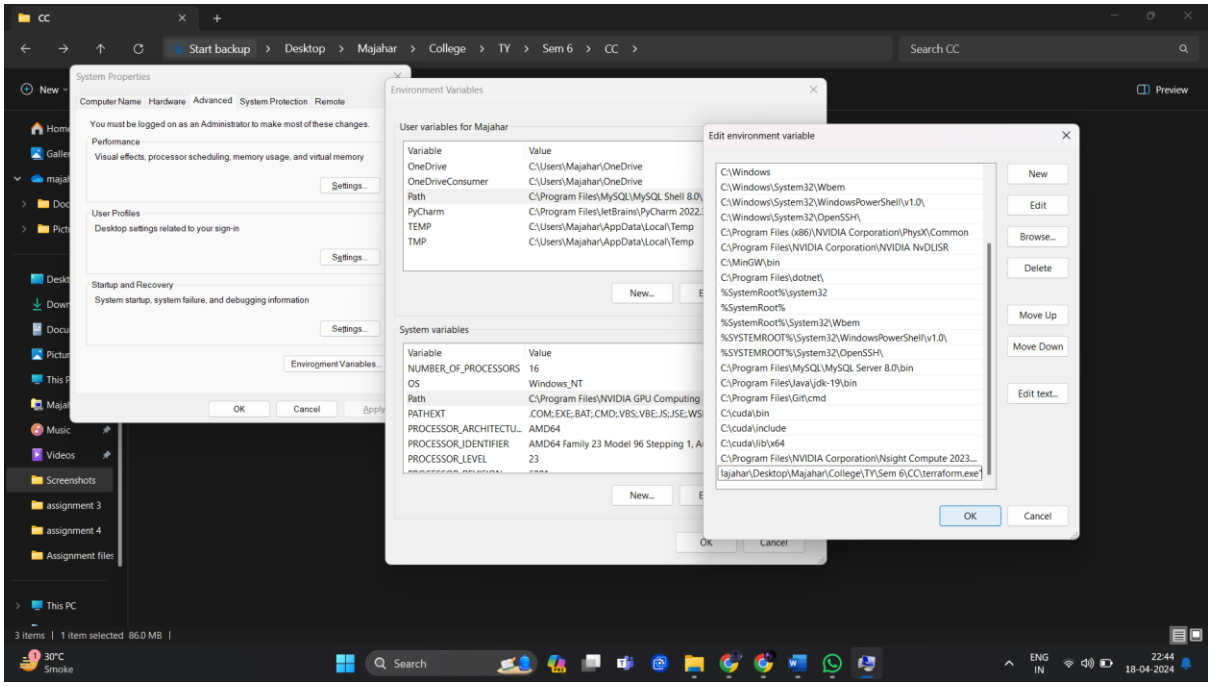


Now that we have got our terraform zip file, unzip in it your required location.

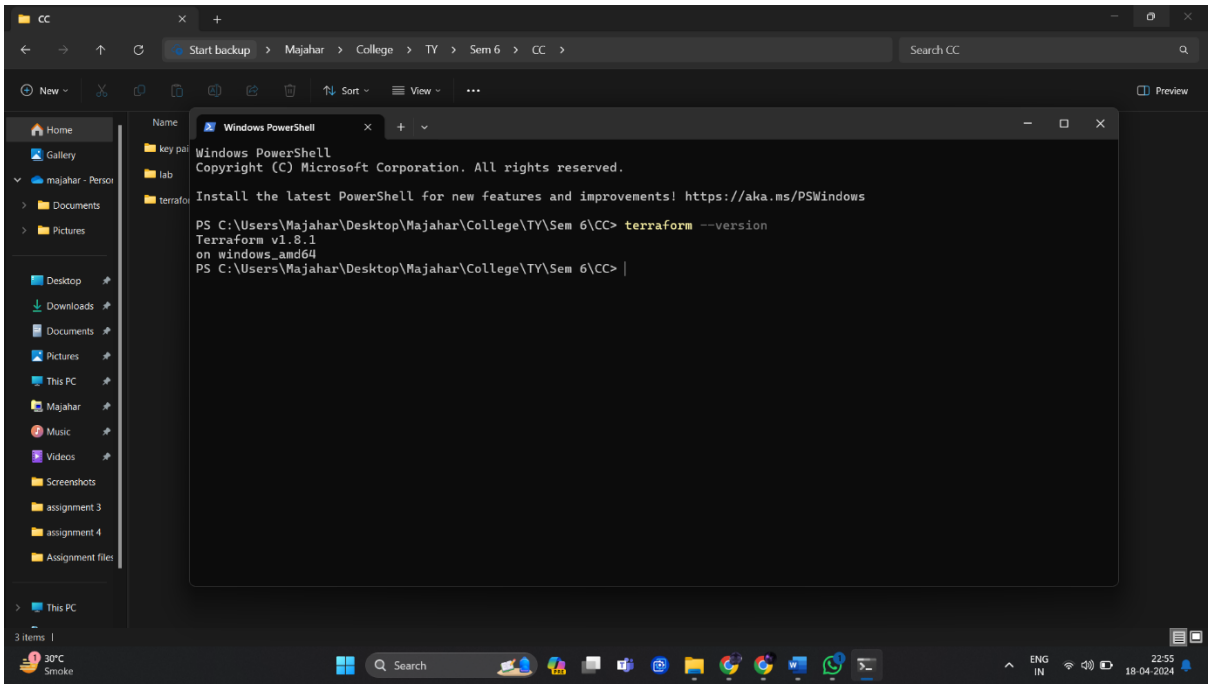

Downloads
This PC > OS (C:) > Users > Majahar > Downloads >
Search Downloads
New
Sort
View
Preview
Name
Date modified
Type
Size
Today
terraform_1.8.1_windows_amd64 18-04-2024 22:37 Compressed (zipp... 26,380 KB
Yesterday
Type: Compressed (zipped) Folder
Size: 25.7 MB
Date modified: 18-04-2024 22:37
p6sid5jg3ydocx 1:01 Microsoft Word D... 16 KB
Earlier this week
s3-bucket.ppk 15-04-2024 22:47 PPK File 2 KB
web 15-04-2024 22:36 Compressed (zipp... 1,259 KB
Last week
css-main (1) 12-04-2024 19:32 Compressed (zipp... 4,329 KB
css-main 12-04-2024 19:11 Compressed (zipp... 4,329 KB
webDeployment.ppk 12-04-2024 18:56 PPK File 2 KB
web (1).pem 12-04-2024 18:34 PEM File 2 KB
web.pem 12-04-2024 18:33 PEM File 2 KB
1_if0YT-YCkYdgr199kZ27g 08-04-2024 10:55 JPG File 24 KB
Uber-Entry-9-by-Anaxid-Croatia 08-04-2024 01:11 JPG File 14 KB
6633677 08-04-2024 01:02 JPG File 90 KB
pngimg.com - tesla_logo_PNG22 08-04-2024 00:58 PNG File 92 KB
download 08-04-2024 00:58 JPEG File 5 KB
OIP 08-04-2024 00:56 JPEG File 11 KB
1_MzbAp6i8xMWx4sRwwxelKw 08-04-2024 00:53 WEBP File 9 KB
389 items
Home
Gallery
majahar - Perso
Documents
Pictures
Desktop
Downloads
Documents
Pictures
This PC
Majahar
Music
Videos
Screenshots
assignment 3
assignment 4
Assignment files
This PC
30°C
Smoke
Search
ENG
IN
22:39
18-04-2024


CC
Start backup > Desktop > Majahar > College > TY > Sem 6 > CC >
Search CC
New
Sort
View
Preview
Name
Date modified
Type
Size
key pair 29-02-2024 11:32 File folder
lab 04-04-2024 09:54 File folder
terraform 17-04-2024 14:55 Application 88,091 KB
3 items
Home
Gallery
majahar - Perso
Documents
Pictures
Desktop
Downloads
Documents
Pictures
This PC
Majahar
Music
Videos
Screenshots
assignment 3
assignment 4
Assignment files
This PC
30°C
Smoke
Search
ENG
IN
22:40
18-04-2024

Now add the application path to system environment variables.



Check if installed.



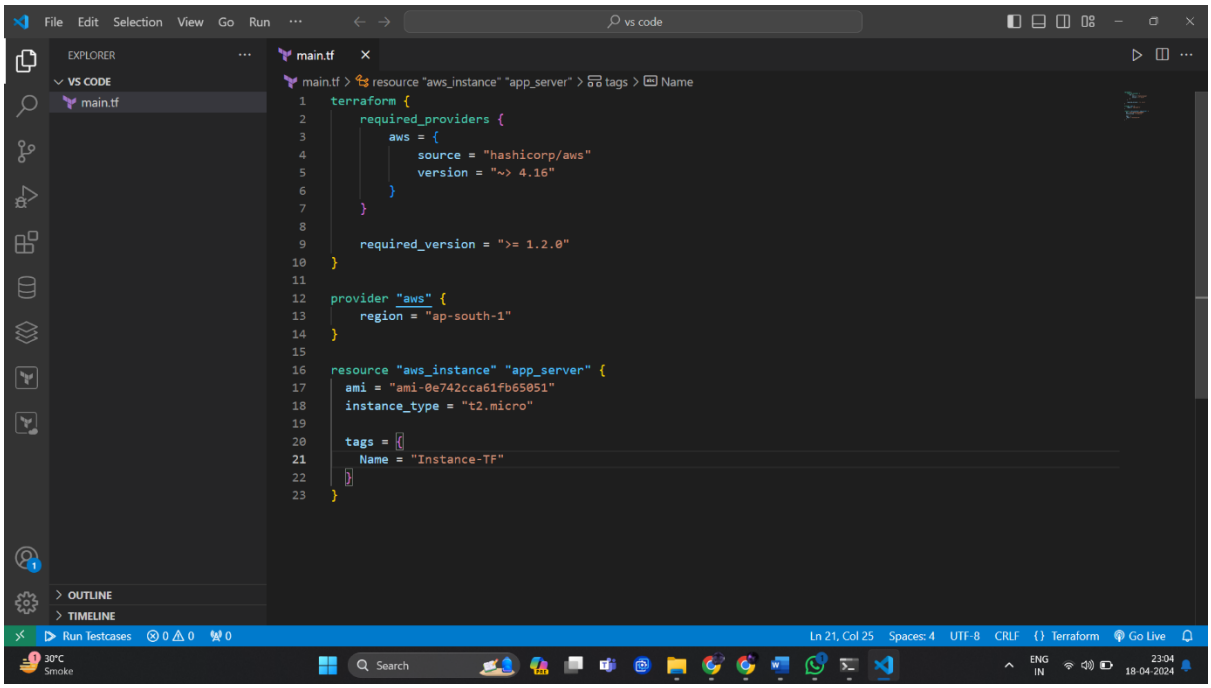
Terraform script to create Infrastructure on any cloud platform

- Now open VS code and create a file with name main.tf

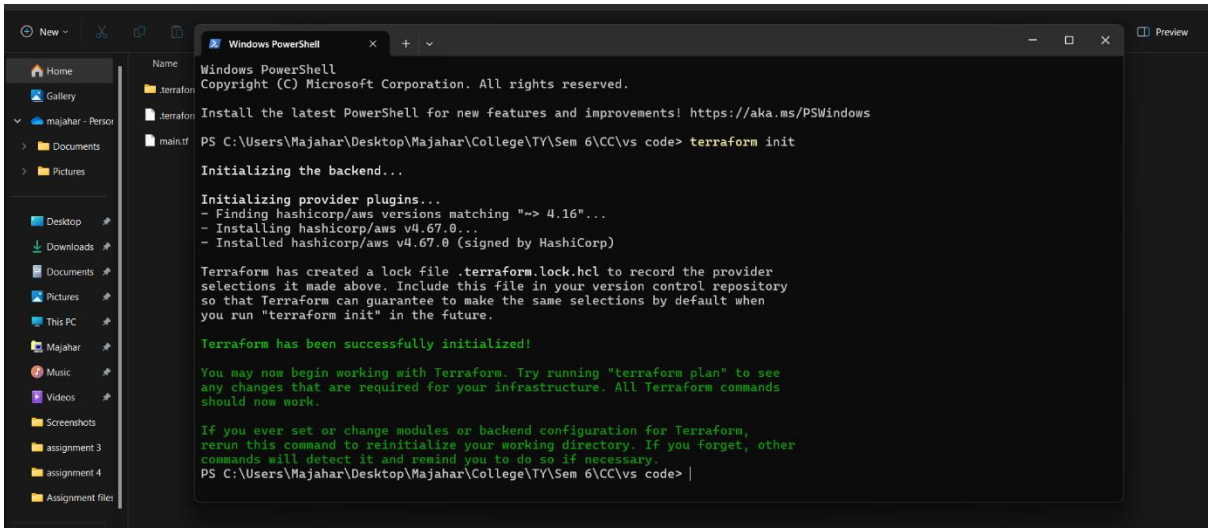
  Mention the provider. We are going to use AWS as our provider.

  The region is Mumbai (ap-south-1)

  The ami id of the VM is added along with other details.



- Using the 'terraform init' command in the folder where we just created out main.tf file to initialize the Terraform Project. This will download any required plugins and initialize the working directory.



- ‘terraform apply’ This will create or update the resources specified in your configuration files after giving the plan.

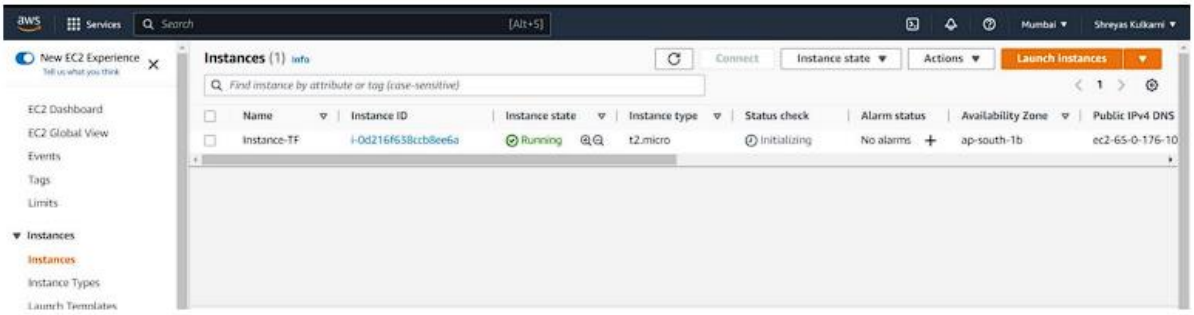
```
D:\TF>terraform apply

Terraform used the selected providers to generate the following execution plan. F
the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.app_server will be created
  + resource "aws_instance" "app_server" {
      + ami                                  = "ami-0e742cca61fb65051"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
```

- The Instance has been deployed Successfully.



We can delete the instance using the terraform destroy. This command will destroy & delete all the instances that were created using terraform.

```
D:\TF>terraform destroy
aws_instance.app_server: Refreshing state... [id=i-0d216f638ccb8ee6a]

Terraform used the selected providers to generate the following execution plan. Res
the following symbols:
  - destroy

Terraform will perform the following actions:

  # aws_instance.app_server will be destroyed
  - resource "aws_instance" "app_server" {
      - ami                                  = "ami-0e742cca61fb65051" -> null
      - arn                                  = "arn:aws:ec2:ap-south-1:905151731728
null
      - associate_public_ip_address          = true -> null
      - availability_zone                    = "ap-south-1b" -> null
      - cpu_core_count                       = 1 -> null
      - cpu_threads_per_core                 = 1 -> null
      - disable_api_stop                     = false -> null
      - disable_api_termination              = false -> null
      - ebs_optimized                        = false -> null
```

```
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_instance.app_server: Destroying... [id=i-0d216f638ccb8ee6a]
aws_instance.app_server: Still destroying... [id=i-0d216f638ccb8ee6a, 10s elapsed]
aws_instance.app_server: Still destroying... [id=i-0d216f638ccb8ee6a, 20s elapsed]
aws_instance.app_server: Destruction complete after 30s

Destroy complete! Resources: 1 destroyed.
```

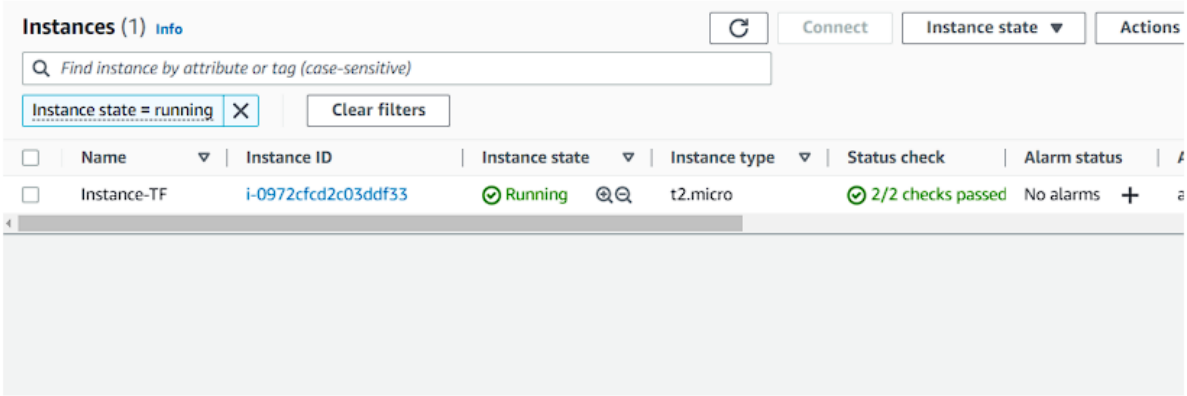
Name	Instance ID	Instance state	Instance type	Status check
Instance-TF	i-0d216f638ccb8ee6a	Terminated	t2.micro	–

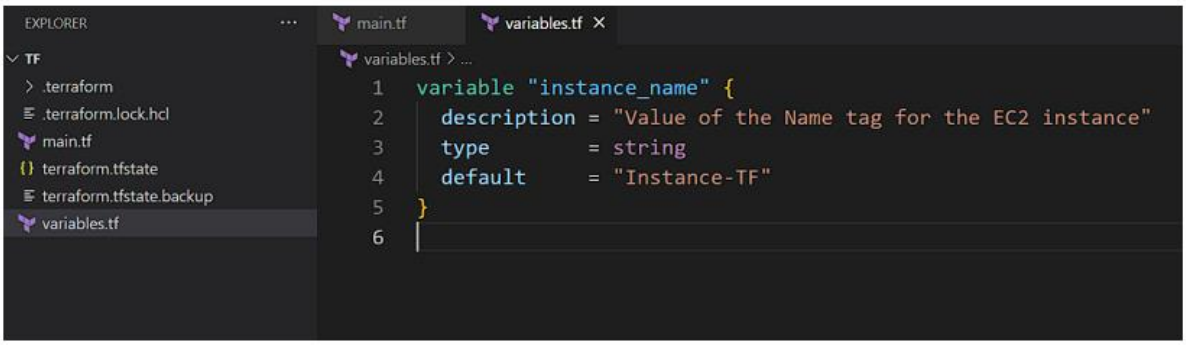
USING INPUT & OUTPUT VARIABLE FILES:

DEFINING INPUT VARIABLES

terraform apply -var "instance_name=Terraform_Instance"

This command will change the instance name by editing the name(It will not delete and create new instance) Terraform configurations can include variables to make your configuration more dynamic and flexible.





Query Data with Outputs: We will use output values to present useful information to the Terraform user.

```
EXPLORER                    main.tf   variables.tf   outputs.tf
TF                          outputs.tf > ...
  .terraform                1  output "instance_id" {
  .terraform.lock.hcl       2    description = "ID of the EC2 instance"
  main.tf                   3    value       = aws_instance.app_server.id
  outputs.tf                4  }
  terraform.tfstate         5
  terraform.tfstate.backup  6  output "instance_public_ip" {
  variables.tf              7    description = "Public IP address of the EC2 instance"
                            8    value       = aws_instance.app_server.public_ip
                            9  }
                           10
```

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

Changes to Outputs:
  + instance_id        = "i-0972cfcd2c03ddf33"
  + instance_public_ip = "3.108.51.231"

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_instance.app_server: Modifying... [id=i-0972cfcd2c03ddf33]
aws_instance.app_server: Modifications complete after 1s [id=i-0972cfcd2c03ddf33]

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

Outputs:

instance_id = "i-0972cfcd2c03ddf33"
instance_public_ip = "3.108.51.231"
```

```
D:\TF>terraform output
instance_id = "i-0972cfcd2c03ddf33"
instance_public_ip = "3.108.51.231"
```

Name	Instance ID	Instance state
Terraform-Instance	i-0972cfcd2c03ddf33	Running

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

Changes to Outputs:
  - instance_id        = "i-0972cfcd2c03ddf33" -> null
  - instance_public_ip = "3.108.51.231" -> null

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_instance.app_server: Destroying... [id=i-0972cfcd2c03ddf33]
aws_instance.app_server: Still destroying... [id=i-0972cfcd2c03ddf33, 10s elapsed]
aws_instance.app_server: Still destroying... [id=i-0972cfcd2c03ddf33, 20s elapsed]
aws_instance.app_server: Still destroying... [id=i-0972cfcd2c03ddf33, 30s elapsed]
aws_instance.app_server: Destruction complete after 40s

Destroy complete! Resources: 1 destroyed.
```

Conclusion:

We have installed terraform and AWS CLI and learned how to use Terraform to create infrastructure as code, specifically an EC2 instance on AWS. We also learned how to use input and output variables to make our code more flexible and reusable. An EC2 Instance was deployed and also destroyed using CLI.

Author – Majahar Kazi

Thursday 18 April 2024 23:55:08 PM IST