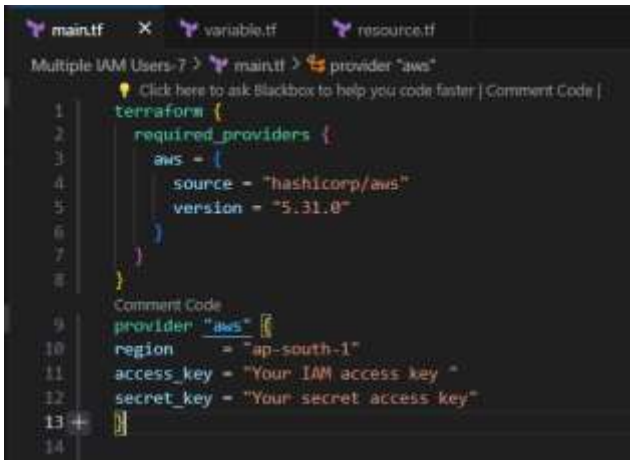
EXPERIMENT – 7

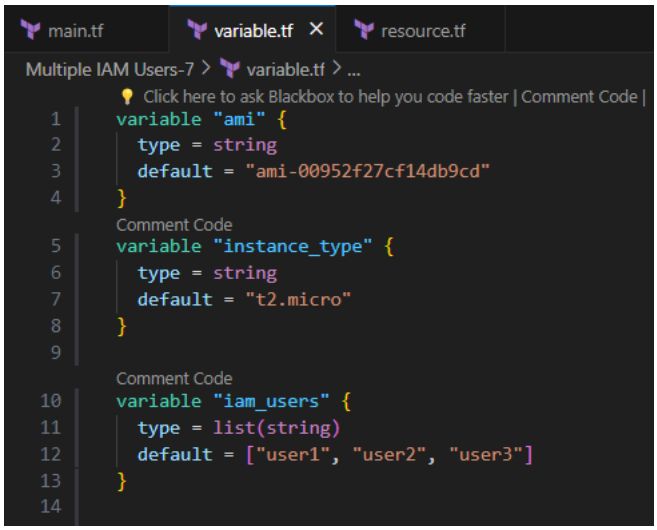
Name: - Shashwat. Dnyaneshwar Kamdi
Batch – 2 [DevOps Non-Hons]
SAP ID- 500092140
Subject – System Provisioning and Configuration Management Lab

Aim: Creating Multiple IAM Users in Terraform.

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



2] Create new file name as “variables.tf”’



3] Create new file name as “resource.tf”’and define a list variable IAM users containing the names of the IAM users that we want to create.

main.tf | variable.tf | resource.tf

Multiple IAM Users-7 > resource.tf > resource "aws_iam_user" "iam_users"

Click here to ask Blackbox to help you code faster | Comment Code |

```
resource "aws_instance" "UPES" {
    ami = var.ami
    instance_type = var.instance_type
    tags = {
        Name = "My-EC2-Instance"
    }
}

Comment Code
resource "aws_iam_user" "iam_users" {
  count = length(var.iam_users)
  name = var.iam_users[count.index]
  tags = {
  Name = "${var.iam_users[count.index]}-user"
  }
}
```

4] Initialize Terraform using command “terraform init”

```
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\Multiple IAM Users-7> 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)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
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.
```

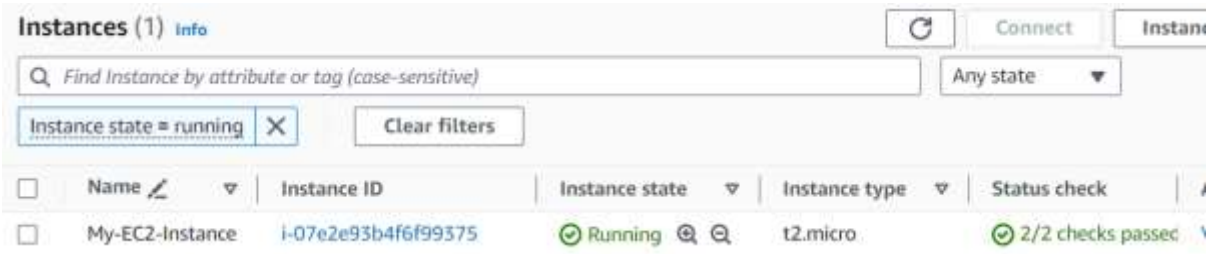
5] Apply it using command “Terraform apply”

```
PS F:\UPES\6th Semester\Sys Provisioning and Cnfg Mgmt\Lab\Terraform-Lab-Scripts\Multiple IAM Users-7> terraform apply -auto-approve

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_iam_user.iam_users[0] will be created
  + resource "aws_iam_user" "iam_users" {
      + arn           = (known after apply)
      + force_destroy = false
      + id            = (known after apply)
      + name          = "user1"
      + path          = "/"
      + tags          = {
          + "Name" = "user1-user"
        }
      + tags_all      = {
          + "Name" = "user1-user"
        }
      + unique_id     = (known after apply)
    }

  # aws_iam_user.iam_users[1] will be created
  + resource "aws_iam_user" "iam_users" {
      + arn           = (known after apply)
      + force_destroy = false
      + id            = (known after apply)
      + name          = "user2"
      + path          = "/"
      + tags          = {
          + "Name" = "user2-user"
        }
      + tags_all      = {
          + "Name" = "user2-user"
        }
      + unique_id     = (known after apply)
    }
```

```
  # aws_iam_user.iam_users[2] will be created
  + resource "aws_iam_user" "iam_users" {
      + arn           = (known after apply)
      + force_destroy = false
      + id            = (known after apply)
      + name          = "user3"
      + path          = "/"
      + tags          = {
          + "Name" = "user3-user"
        }
      + tags_all      = {
          + "Name" = "user3-user"
        }
      + unique_id     = (known after apply)
    }

  # aws_instance.UPES will be created
  + resource "aws_instance" "UPES" {
      + ami                                  = "ami-00952f27cf14db9cd"
      + 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_lifecycle                   = (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)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "My-EC2-Instance"
        }
      + tags_all                             = {
          + "Name" = "My-EC2-Instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)
    }

Plan: 4 to add, 0 to change, 0 to destroy.
aws_iam_user.iam_users[2]: Creating...
]
aws_iam_user.iam_users[0]: Creation complete after 2s [id=user1]
aws_iam_user.iam_users[1]: Creation complete after 2s [id=user2]
aws_instance.UPES: Still creating... [10s elapsed]
aws_instance.UPES: Still creating... [20s elapsed]
aws_instance.UPES: Creation complete after 22s [id=i-07e2e93b4f6f99375]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.
```

6] Verify Resources on AWS Management Console.

EC2



User