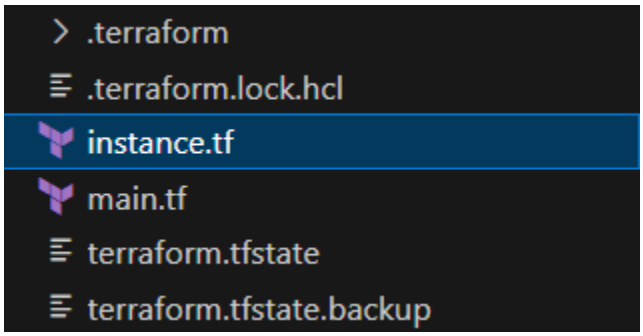
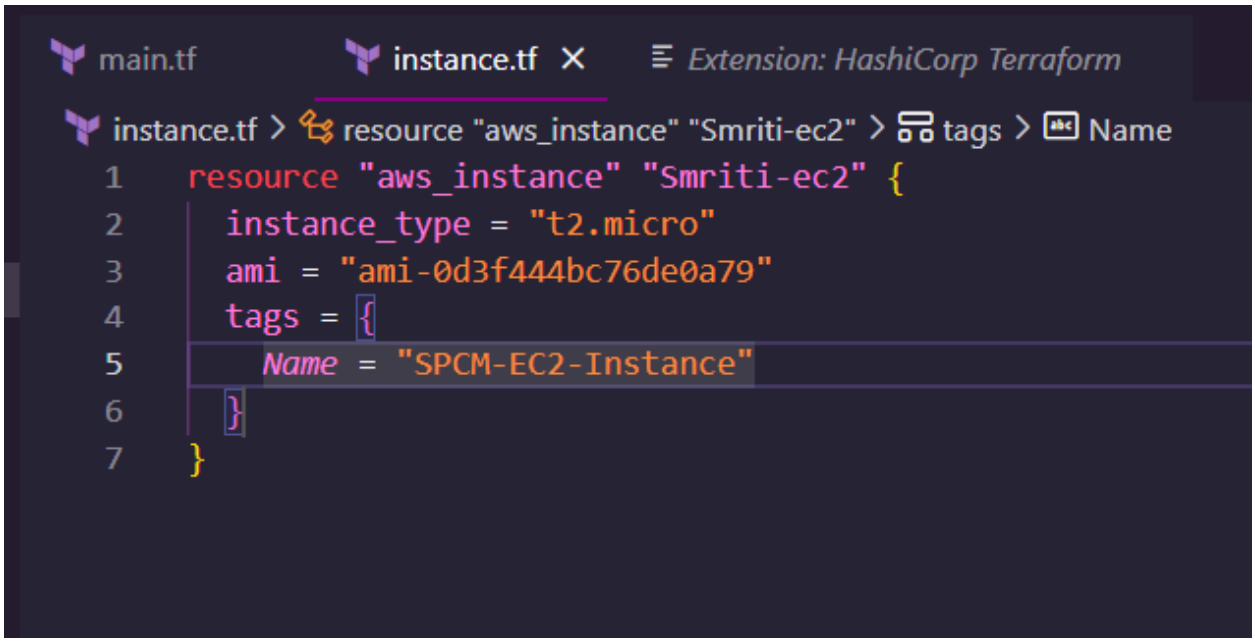
# EXPERIMENT 3:

# Provisioning an EC2 Instance on AWS

1. Create a terraform configuration file for EC2 instance called ***instance.tf***




2. Note down the ami code from your AWS account.
3. Write down the following content in instance.tf




4. Review plan - Run the following command to see what terraform will do.

```
D:\docss\UPES\sem 6\SPCM Lab>terraform plan

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_instance.Smriti-ec2 will be created
  + resource "aws_instance" "Smriti-ec2" {
      + ami                                  = "ami-0d3f444bc76de0a79"
      + 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)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
```

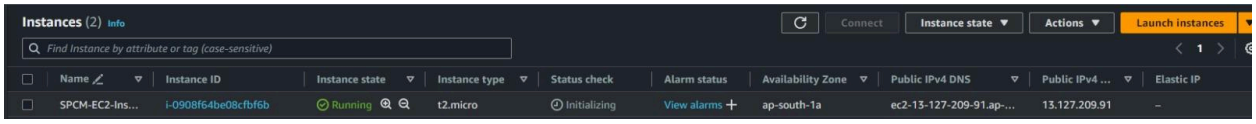
```
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "SPCM-EC2-Instance"
        }
      + tags_all                             = {
          + "Name" = "SPCM-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: 1 to add, 0 to change, 0 to destroy.

---

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
you run "terraform apply" now.

D:\docss\UPES\sem 6\SPCM Lab>
```

5. Apply changes - Apply the changes to create AWS resources

```
D:\docss\UPES\sem 6\SPCM Lab>terraform apply

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_instance.Smriti-ec2 will be created
  + resource "aws_instance" "Smriti-ec2" {
      + ami                                  = "ami-0d3f444bc76de0a79"
      + 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)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
```

```
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "SPCM-EC2-Instance"
        }
      + tags_all                             = {
          + "Name" = "SPCM-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: 1 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_instance.Smriti-ec2: Creating...
aws_instance.Smriti-ec2: Still creating... [10s elapsed]
aws_instance.Smriti-ec2: Still creating... [20s elapsed]
aws_instance.Smriti-ec2: Still creating... [30s elapsed]
aws_instance.Smriti-ec2: Creation complete after 33s [id=i-0908f64be08cfbf6b]

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

D:\docss\UPES\sem 6\SPCM Lab>
```

6. Verify by logging into your AWS account and check if your resources is created.




7. When you are done with experimenting, run the following command to destroy the created resources.

```
D:\docss\UPES\sem 6\SPCM Lab>terraform destroy
aws_instance.Smriti-ec2: Refreshing state... [id=i-0908f64be08cfbf6b]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  - destroy

Terraform will perform the following actions:

  # aws_instance.Smriti-ec2 will be destroyed
  - resource "aws_instance" "Smriti-ec2" {
      - ami                                  = "ami-0d3f444bc76de0a79" -> null
      - arn                                  = "arn:aws:ec2:ap-south-1:774931074009:instance/i-0908f64be08cfbf6b" -> null
      - associate_public_ip_address          = true -> null
      - availability_zone                    = "ap-south-1a" -> 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
      - get_password_data                    = false -> null
      - hibernation                          = false -> null
      - id                                   = "i-0908f64be08cfbf6b" -> null
      - instance_initiated_shutdown_behavior = "stop" -> null
      - instance_state                       = "running" -> null
      - instance_type                        = "t2.micro" -> null
      - ipv6_address_count                   = 0 -> null
      - ipv6_addresses                       = [] -> null
      - monitoring                           = false -> null
      - placement_partition_number           = 0 -> null
      - primary_network_interface_id         = "eni-0b5e477858c21e5d3" -> null
      - private_dns                          = "ip-172-31-46-17.ap-south-1.compute.internal" -> null
      - private_ip                           = "172.31.46.17" -> null
      - public_dns                           = "ec2-13-127-209-91.ap-south-1.compute.amazonaws.com" -> null
      - public_ip                            = "13.127.209.91" -> null
      - secondary_private_ips                = [] -> null
      - security_groups                      = [
          - "default",
        ] -> null
      - source_dest_check                    = true -> null

      - private_dns_name_options {
          - enable_resource_name_dns_a_record    = false -> null
          - enable_resource_name_dns_aaaa_record = false -> null
          - hostname_type                        = "ip-name" -> null
        }

      - root_block_device {
          - delete_on_termination = true -> null
          - device_name           = "/dev/xvda" -> null
          - encrypted             = false -> null
          - iops                  = 3000 -> null
          - tags                  = {} -> null
          - throughput            = 125 -> null
          - volume_id             = "vol-03c76fb607b72f13e" -> null
          - volume_size           = 8 -> null
          - volume_type           = "gp3" -> null
        }
    }

Plan: 0 to add, 0 to change, 1 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_instance.Smriti-ec2: Destroying... [id=i-0908f64be08cfbf6b]
aws_instance.Smriti-ec2: Still destroying... [id=i-0908f64be08cfbf6b, 10s elapsed]
aws_instance.Smriti-ec2: Still destroying... [id=i-0908f64be08cfbf6b, 20s elapsed]
aws_instance.Smriti-ec2: Still destroying... [id=i-0908f64be08cfbf6b, 30s elapsed]
aws_instance.Smriti-ec2: Still destroying... [id=i-0908f64be08cfbf6b, 40s elapsed]
aws_instance.Smriti-ec2: Destruction complete after 40s

Destroy complete! Resources: 1 destroyed.

D:\docss\UPES\sem 6\SPCM Lab>
```