SPCM - LAB

6th Sem

Submitted To:

Dr. Hitesh Kumar Sharma

School of Computer Science

Submitted By:

Kushagra Singh

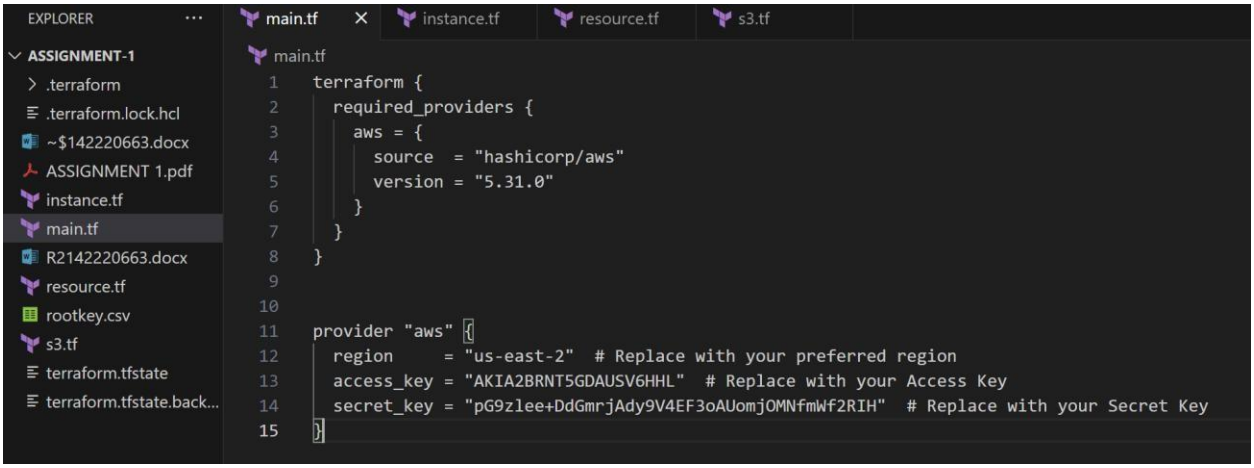
R2142228077

500107601

Batch 2-NH

Writing Terraform Scripts to perform the following task 2 ec2 Instances, VPN and S3

main.tf



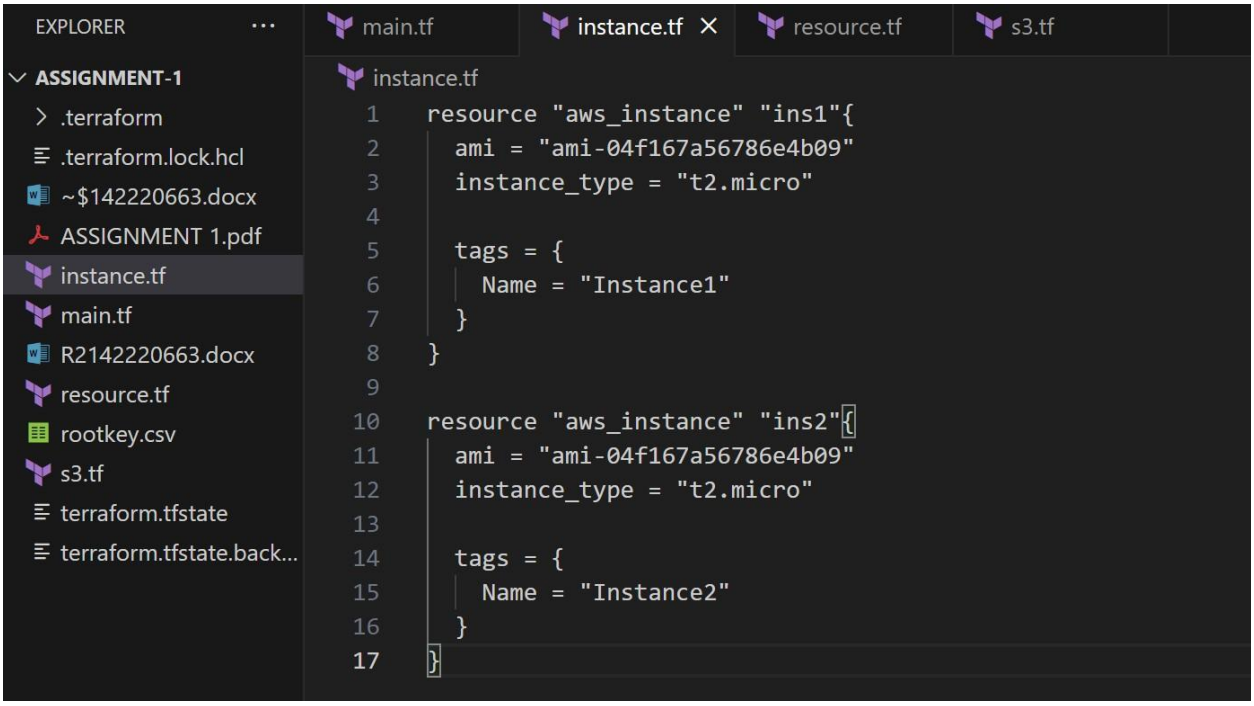
Running **terraform init**

Terraform init to initialize the

```
D_DATA|Kushagra\UPES COLLEGE STUDY MATERIAL\SEM6\SPCM\labAssigment-1
ing the backend...
ing provider plugins...
ing hashicorp/aws versions '>3.31.0'...
alling hashicorp/aws v3.31.0...
oading hashicorp/aws 3.31.0 (signed by HashiCorp)
orm has created a lock file .terraform.lock.hcl to record provider
ions it made above. Include this file in your version control repository
 Terraform can guarantee to make the same selections by default when
'terraform init' in the future.

orm has been successfully initialized!
 now begin working with Terraform, Iry running 'terraform plan' to see any
 that are required for your infrastructure. All Terraform commands should
k.
 ever set or change modules or backend configuration for Terraform,
ID_DATA|Kushagra\UPES COLLEGE STUDY MATERIAL\SEM6\SPCM\labAssigment-1>
```

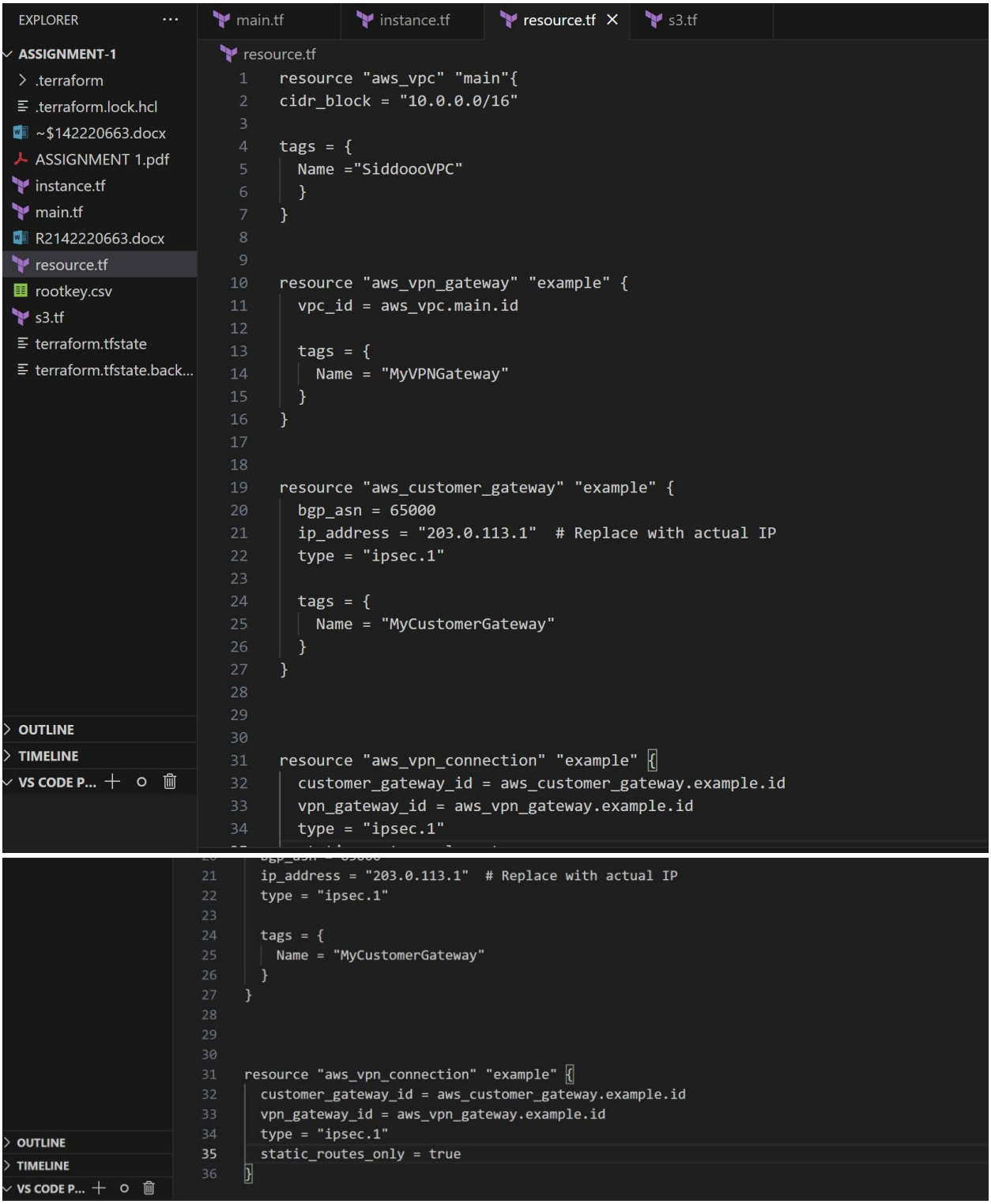
terraform folder which will have the aws provider plugin installed

instance.tf



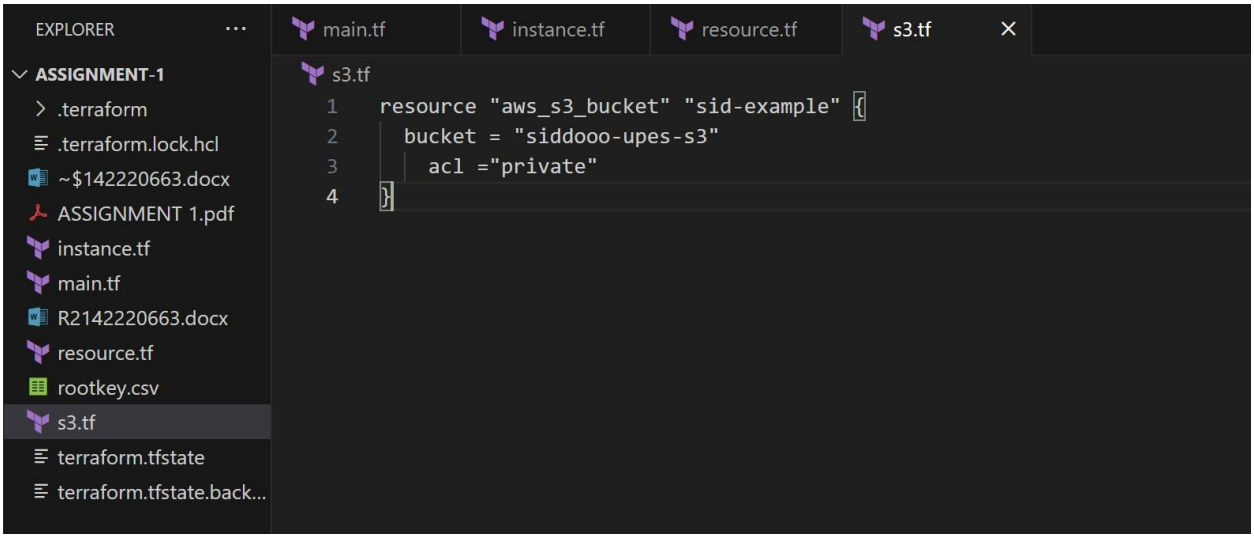
This file holds the iac code to make 2 instances - t2-micro ec2 machines

resource.tf



This resource.tf hold the iac code to create vpc , the customer gateway and the vpn connection.

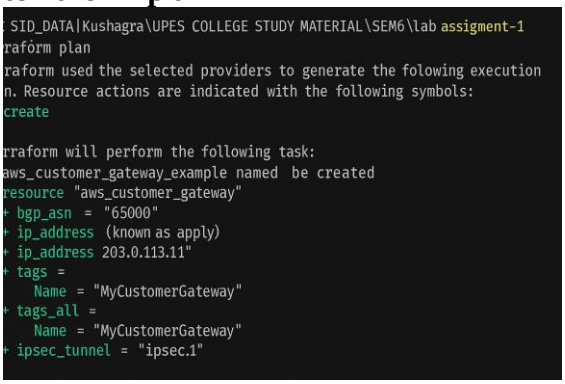
s3.tf

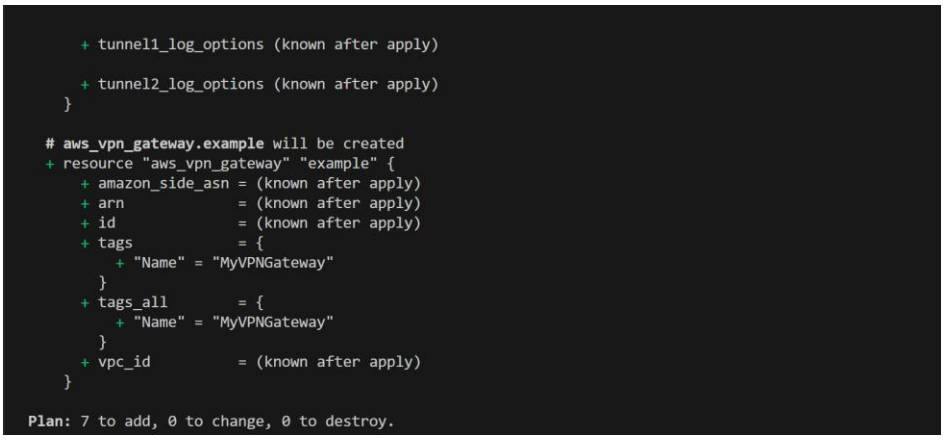


The s3.tf hold the code to create a s3 bucket which has a unique name

Outputs:

terraform plan

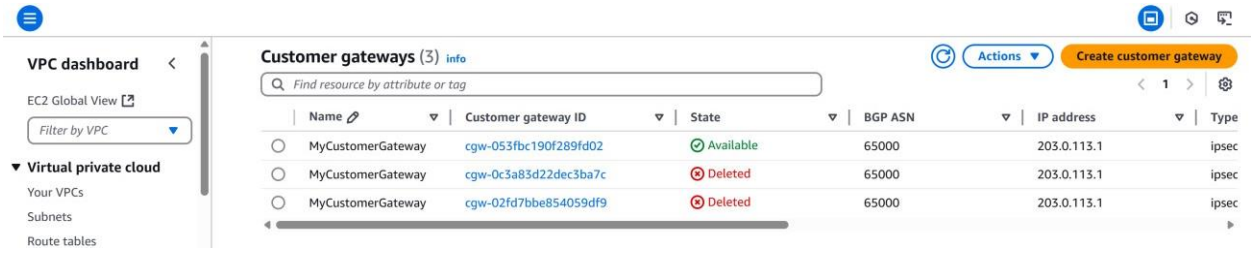




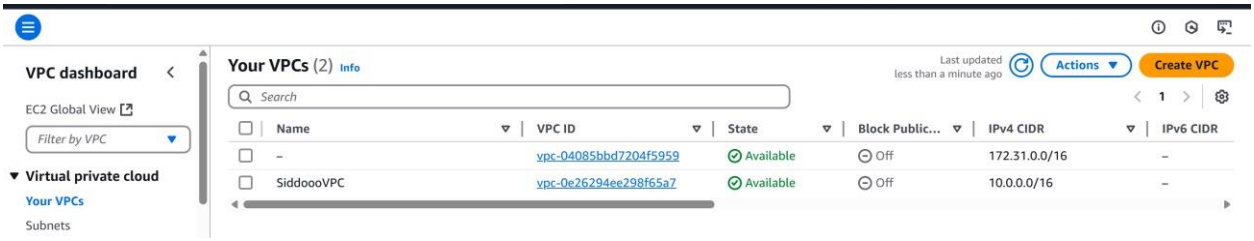
Terraform apply

```
PS C:\Kushagra_DATASKUSHAGRA\UPES COLLEGE STUDY MATERIAL\SEM6\SPCM\lab\Assignment-1> 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_customer_gateway.example will be created
+ resource "aws_customer_gateway" "example" {
    arm = (known after apply)
    bgp_asm = "65800"
    id = (known after apply)
    ip_address = "283.0.113.1"
    tags = (
    + "Name" = "MyCustomerGateway"
    )
    tags_all = (
    + "Name" = "MyCustomerGateway"
    )
    type = "ipsec.1"
}

# aws_instance.insl will be created
+ resource "aws_instance" "insl" {
    arm = "ami-04f167a56786e4b69"
    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

3: acl ="private"

Use the aws_s3_bucket_acl resource instead

(and 2 more similar warnings elsewhere)

(and 2 more similar warnings elsewhere)

(and 2 more similar warnings elsewhere)

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
PS C:\Kushagra_DATASKUSHAGRA\UPES COLLEGE STUDY MATERIAL\SEM6\SPCM\lab\Assignment-1>
```

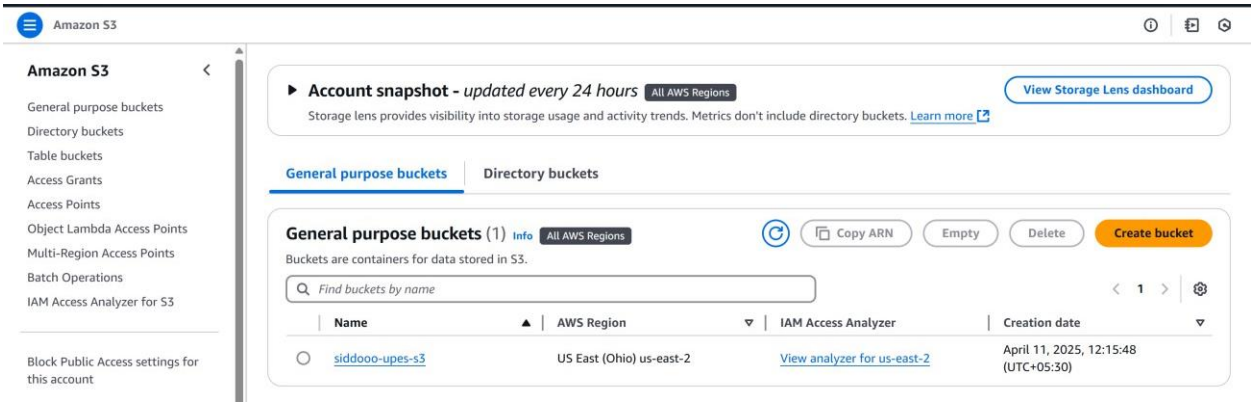
Customer Gateway



VPC



S3



Instances