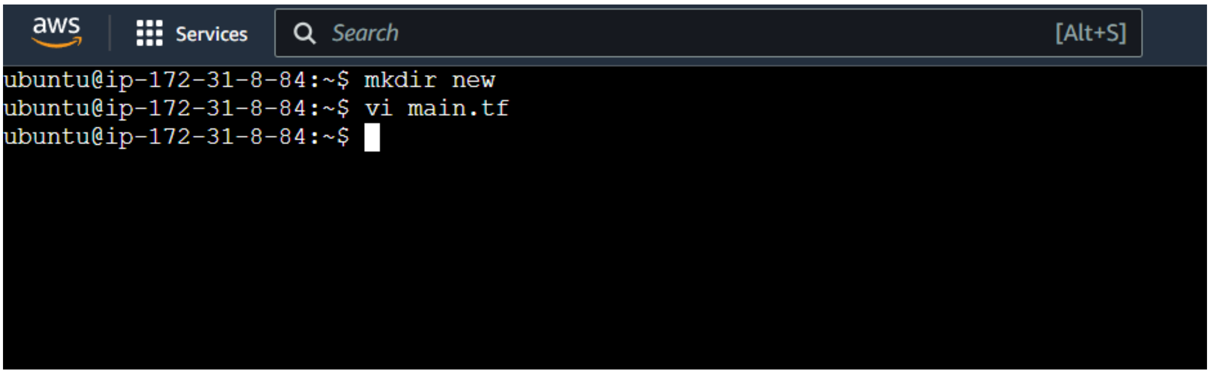
Task – 17

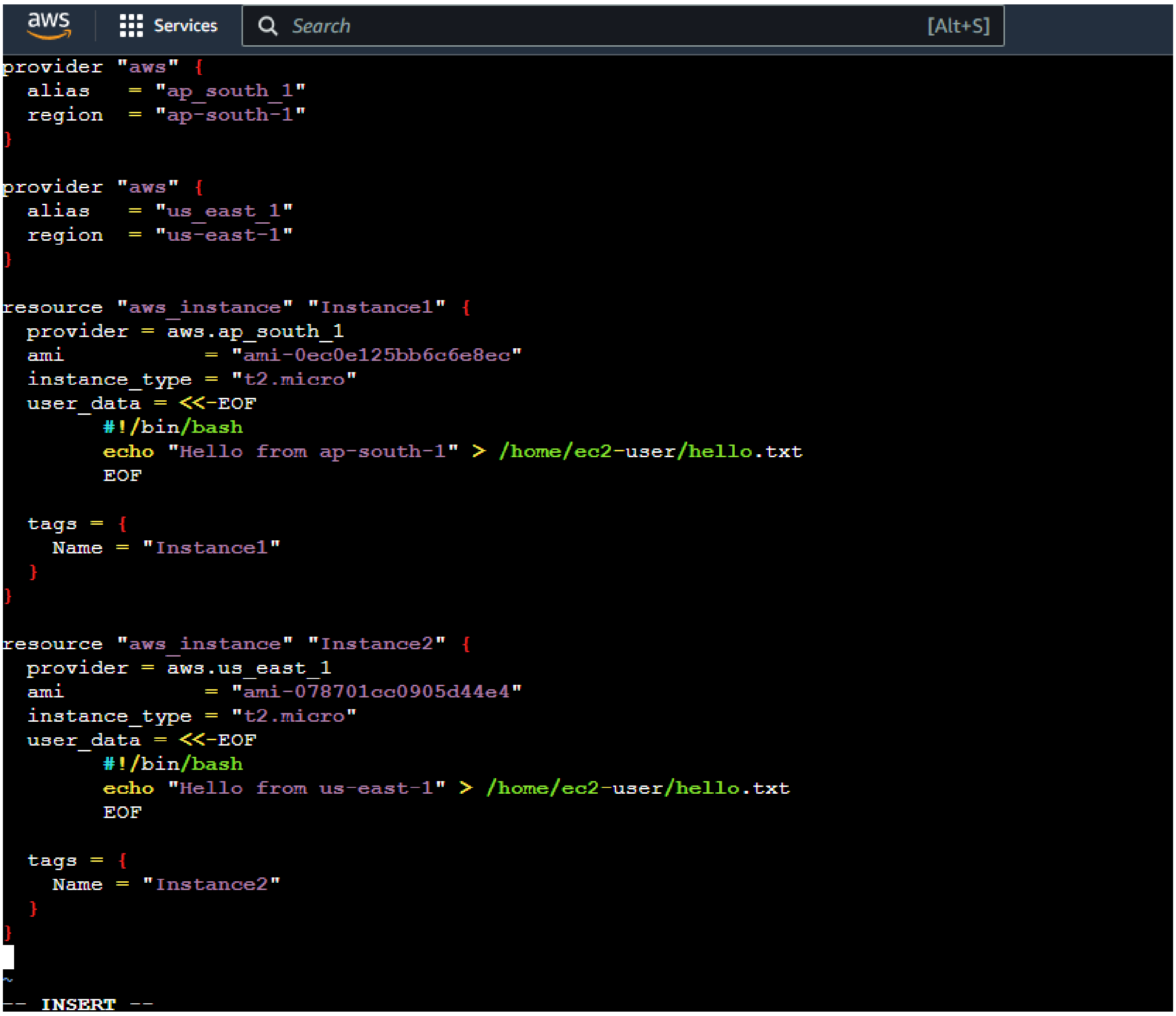
Launch Linux EC2 instances in two regions using a single Terraform file

Creating a new directory:



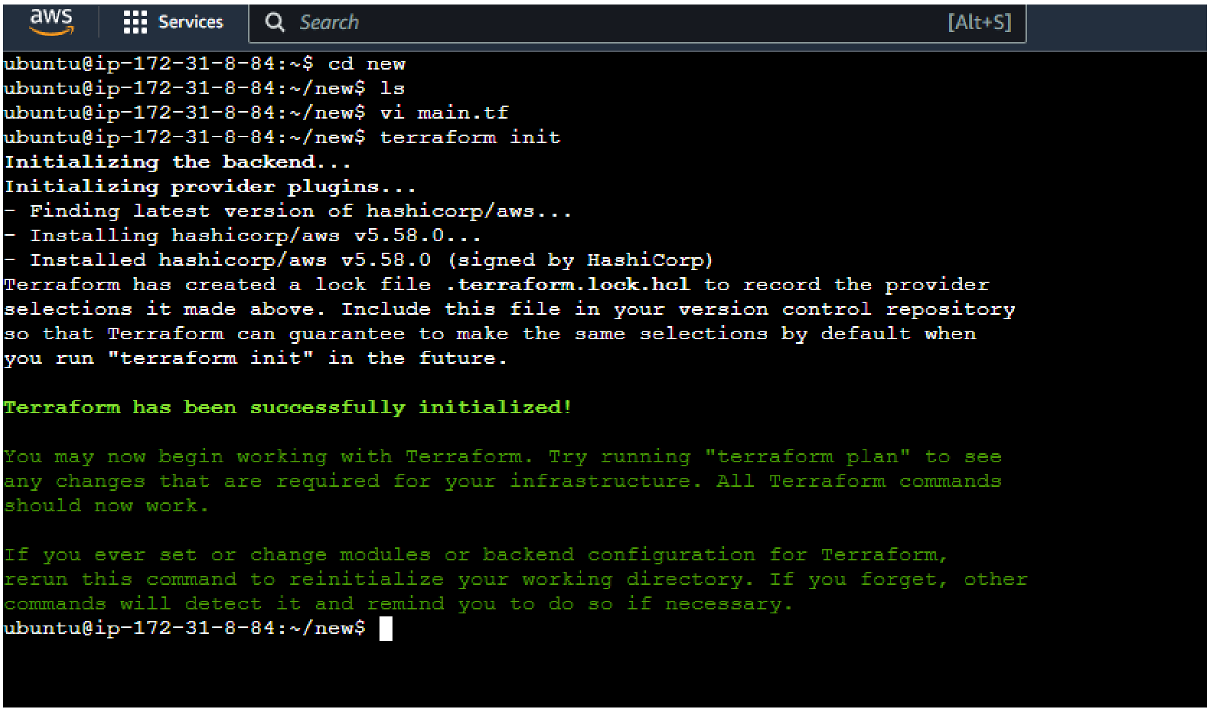
```
ubuntu@ip-172-31-8-84:~$ mkdir new
ubuntu@ip-172-31-8-84:~$ vi main.tf
ubuntu@ip-172-31-8-84:~$
```

Writing Terraform file:

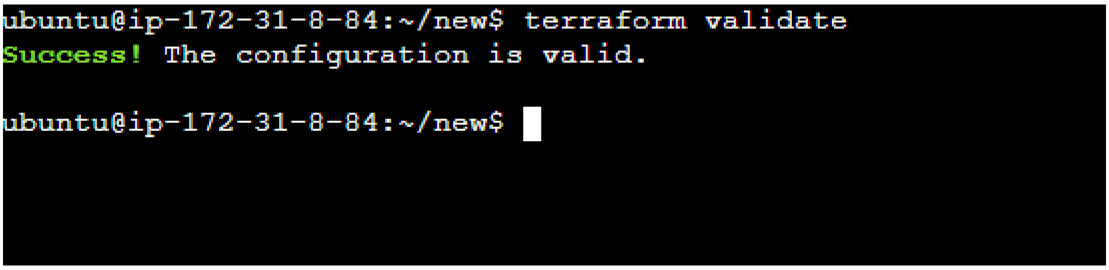


```
provider "aws" {
  alias   = "ap_south_1"
  region  = "ap-south-1"
}

provider "aws" {
  alias   = "us_east_1"
  region  = "us-east-1"
}

resource "aws_instance" "Instance1" {
  provider = aws.ap_south_1
  ami           = "ami-0ec0e125bb6c6e8ec"
  instance_type = "t2.micro"
  user_data = <<-EOF
        #!/bin/bash
        echo "Hello from ap-south-1" > /home/ec2-user/hello.txt
        EOF

  tags = {
    Name = "Instance1"
  }
}

resource "aws_instance" "Instance2" {
  provider = aws.us_east_1
  ami           = "ami-078701cc0905d44e4"
  instance_type = "t2.micro"
  user_data = <<-EOF
        #!/bin/bash
        echo "Hello from us-east-1" > /home/ec2-user/hello.txt
        EOF

  tags = {
    Name = "Instance2"
  }
}

~
-- INSERT --
```

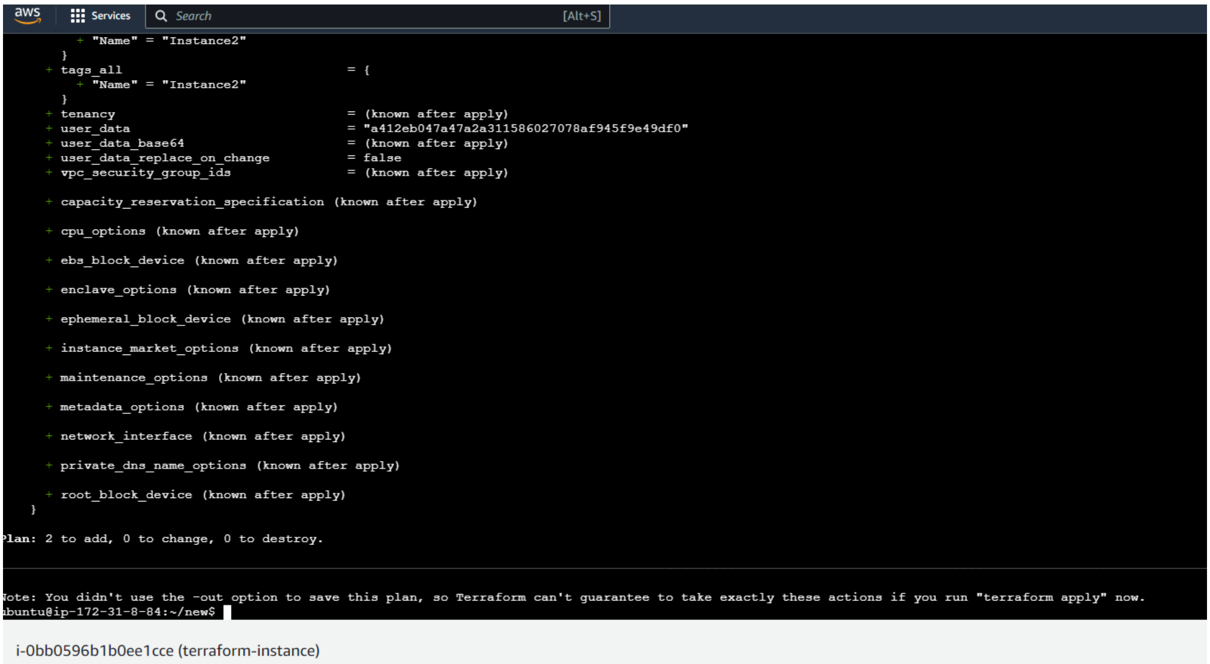
Initializing and executing terraform file:



```
ubuntu@ip-172-31-8-84:~$ cd new
ubuntu@ip-172-31-8-84:~/new$ ls
ubuntu@ip-172-31-8-84:~/new$ vi main.tf
ubuntu@ip-172-31-8-84:~/new$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.58.0...
- Installed hashicorp/aws v5.58.0 (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.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
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.
ubuntu@ip-172-31-8-84:~/new$
```

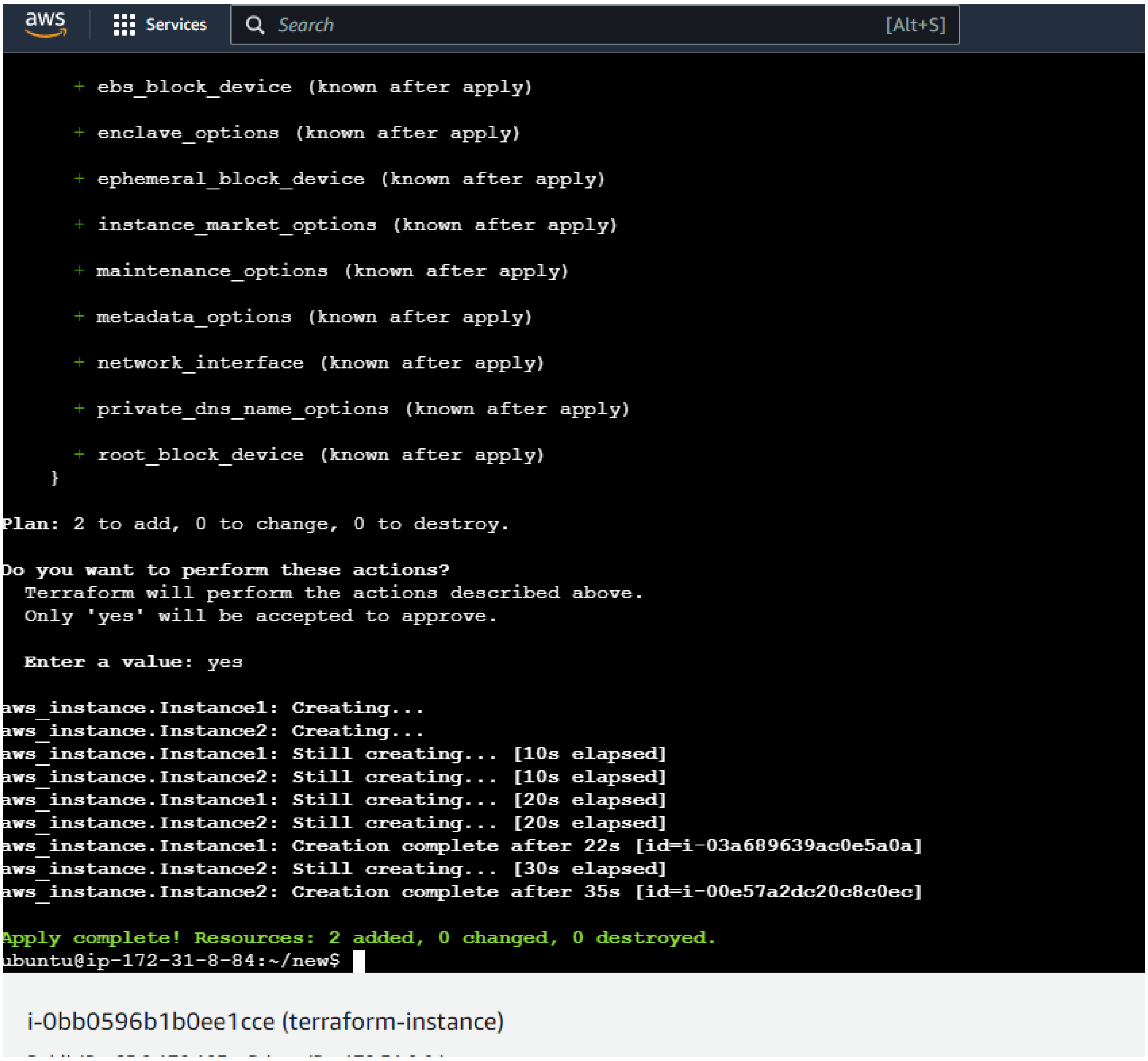


```
ubuntu@ip-172-31-8-84:~/new$ terraform validate
Success! The configuration is valid.

ubuntu@ip-172-31-8-84:~/new$
```



```
          + "Name" = "Instance2"
        }
      + tags_all                             = {
          + "Name" = "Instance2"
        }
      + tenancy                              = (known after apply)
      + user_data                            = "a412eb047a47a2a311586027078af945f9e49df0"
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

Plan: 2 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.
ubuntu@ip-172-31-8-84:~/new$
```

i-0bb0596b1b0ee1cce (terraform-instance)

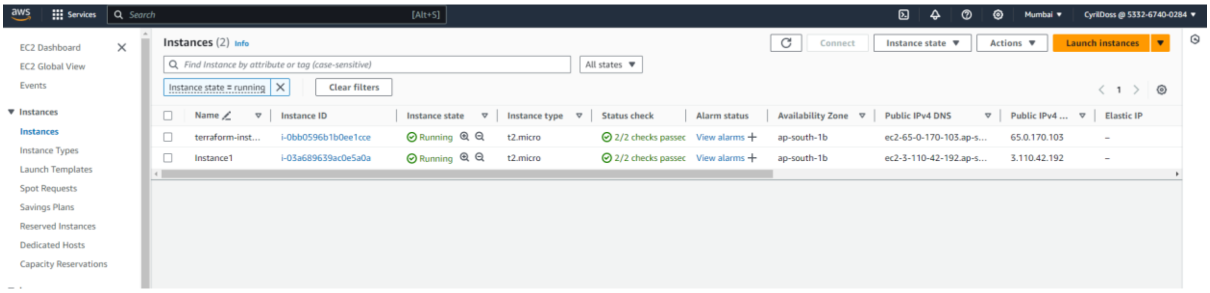


```
      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

Plan: 2 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.Instance1: Creating...
aws_instance.Instance2: Creating...
aws_instance.Instance1: Still creating... [10s elapsed]
aws_instance.Instance2: Still creating... [10s elapsed]
aws_instance.Instance1: Still creating... [20s elapsed]
aws_instance.Instance2: Still creating... [20s elapsed]
aws_instance.Instance1: Creation complete after 22s [id=i-03a689639ac0e5a0a]
aws_instance.Instance2: Still creating... [30s elapsed]
aws_instance.Instance2: Creation complete after 35s [id=i-00e57a2dc20c8c0ec]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-8-84:~/new$
```

i-0bb0596b1b0ee1cce (terraform-instance)

Checking Whether instance has been created

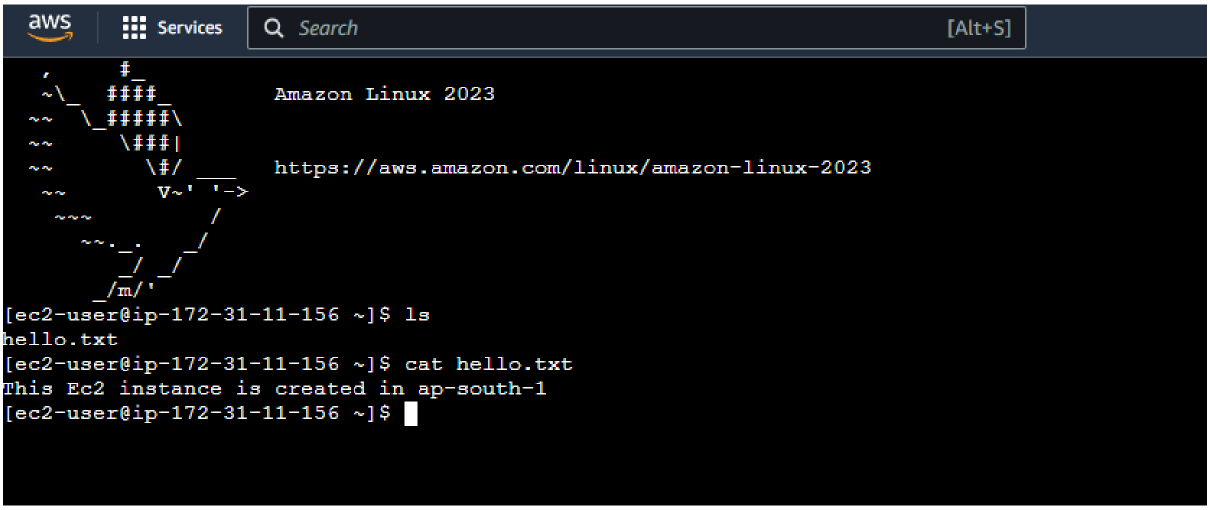
Instance 1:



Instances (2) Info

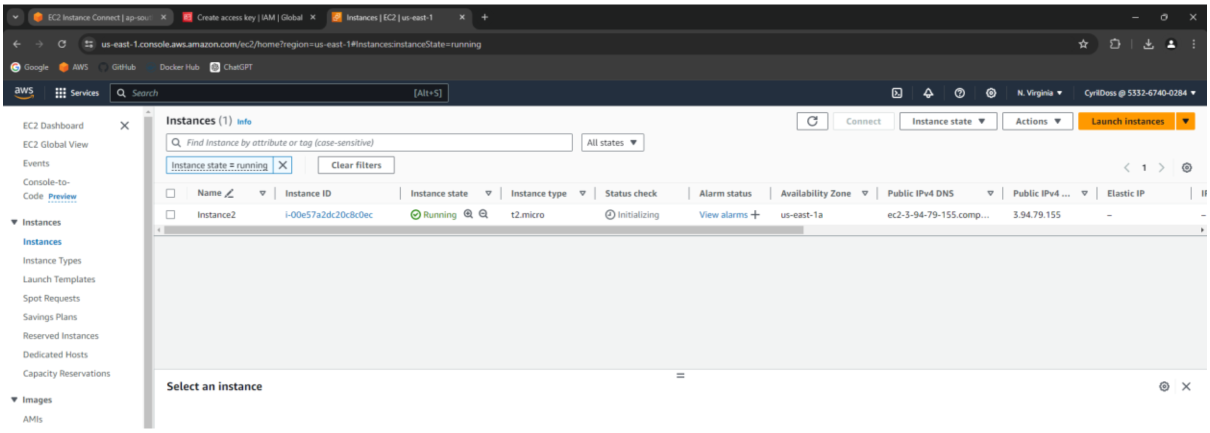
Instance state = running

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
□	terraform-inst...	i-0bb0596b1b0ee1cce	Running	t2.micro	2/2 checks passec	View alarms +	ap-south-1b	ec2-65-0-170-103.ap-s...	65.0.170.103	–
□	Instance1	i-03a689639ac0e5a0a	Running	t2.micro	2/2 checks passec	View alarms +	ap-south-1b	ec2-3-110-42-192.ap-s...	3.110.42.192	–



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-11-156 ~]$ ls
hello.txt
[ec2-user@ip-172-31-11-156 ~]$ cat hello.txt
This Ec2 instance is created in ap-south-1
[ec2-user@ip-172-31-11-156 ~]$ 
```

Instance 2:



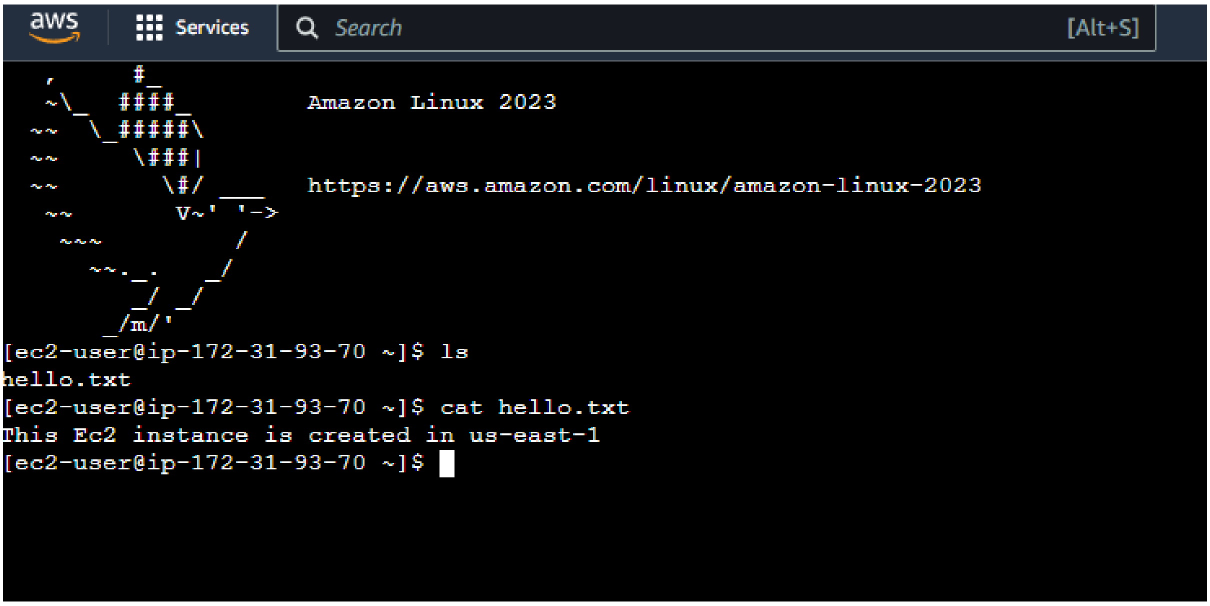
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:instanceState=running

Instances (1) Info

Instance state = running

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IF
□	Instance2	i-00e57a2dc20c8c0ec	Running	t2.micro	Initializing	View alarms +	us-east-1a	ec2-3-94-79-155.comp...	3.94.79.155	–	–

Select an instance



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-93-70 ~]$ ls
hello.txt
[ec2-user@ip-172-31-93-70 ~]$ cat hello.txt
This Ec2 instance is created in us-east-1
[ec2-user@ip-172-31-93-70 ~]$
```