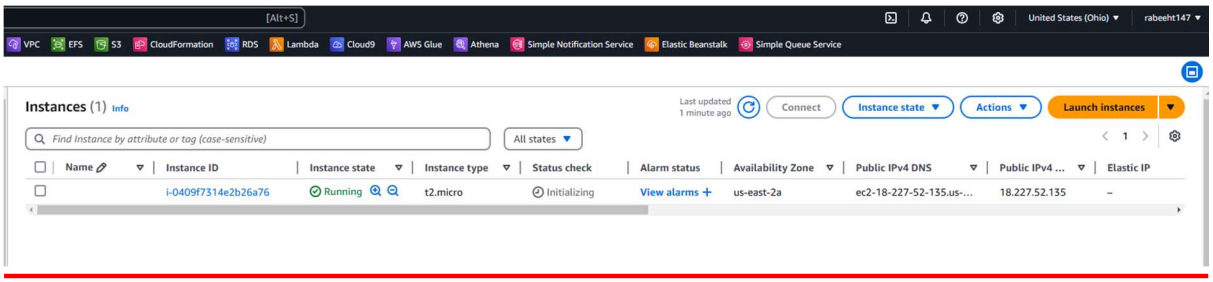
Terraform Assignment – 1

```
provider "aws" {
   region = "us-east-2"
   access_key = "AKIA4MTWGW7C33ESVYP3"
   secret_key  = "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"
   }

data "aws_vpc" "default" {
   default = true
   }

resource "aws_instance" "ubuntu" {
   ami = "ami-04b4f1a9cf54c11d0"
   instance_type = "t2.micro"
   }

```

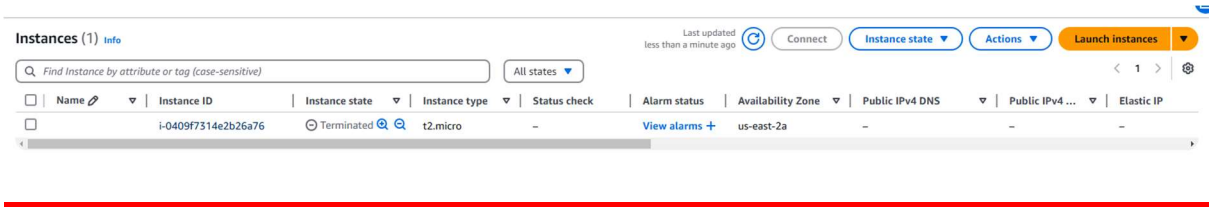
```
user@user:~/terra$ nano demo.tf
user@user:~/terra$ terraform apply
data.aws_vpc.default: Reading...
data.aws_vpc.default: Read complete after 3s [id=vpc-0cc3949c9cf957a84]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the foll
  + create

Terraform will perform the following actions:

  # aws_instance.ubuntu will be created
  + resource "aws_instance" "ubuntu" {
      + ami                                  = "ami-0884d2865dbe9de4b"
      + 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)
      + enable_primary_ipv6                  = (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)
```

```
      + 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: 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.ubuntu: Creating...
aws_instance.ubuntu: Still creating... [10s elapsed]
aws_instance.ubuntu: Creation complete after 18s [id=i-0409f7314e2b26a76]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
user@user:~/terra$
```



```
user@user:~/terra$ terraform destroy
data.aws_vpc.default: Reading...
aws_instance.ubuntu: Refreshing state... [id=i-0409f7314e2b26a76]
data.aws_vpc.default: Read complete after 3s [id=vpc-0cc3949c9cf957a84]

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.ubuntu will be destroyed
  - resource "aws_instance" "ubuntu" {
      - ami                                  = "ami-0884d2865dbe9de4b" -> null
      - arn                                  = "arn:aws:ec2:us-east-2:851725170629:instance/i-0409f7314e2b26a76" -> null
      - associate_public_ip_address          = true -> null
      - availability_zone                    = "us-east-2a" -> 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-0409f7314e2b26a76" -> 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-05f55890ef5dbcd93" -> null
      - private_dns                          = "ip-172-31-2-224.us-east-2.compute.internal" -> null
      - private_ip                           = "172.31.2.224" -> null
      - public_dns                           = "ec2-18-227-52-135.us-east-2.compute.amazonaws.com" -> null
      - public_ip                            = "18.227.52.135" -> null
      - secondary_private_ips                = [] -> null
      - security_groups                      = [
          - "default",
        ] -> null
      - source_dest_check                    = true -> null
      - subnet_id                            = "subnet-0814016888677d3d0" -> null
```

```
      - root_block_device {
          - delete_on_termination = true -> null
          - device_name           = "/dev/sda1" -> null
          - encrypted             = false -> null
          - iops                  = 100 -> null
          - tags                  = {} -> null
          - tags_all              = {} -> null
          - throughput            = 0 -> null
          - volume_id             = "vol-0e36b8546f8761cc9" -> null
          - volume_size           = 8 -> null
          - volume_type           = "gp2" -> null
            # (1 unchanged attribute hidden)
        }
    }

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.ubuntu: Destroying... [id=i-0409f7314e2b26a76]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 10s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 20s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 30s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 40s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 50s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 1m0s elapsed]
aws_instance.ubuntu: Still destroying... [id=i-0409f7314e2b26a76, 1m10s elapsed]
aws_instance.ubuntu: Destruction complete after 1m15s

Destroy complete! Resources: 1 destroyed.
user@user:~/terra$
```


Instances (1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Public IPv4 ...
Elastic IP
i-0409f7314e2b26a76
Terminated
t2.micro
–
View alarms
us-east-2a
–
–
–