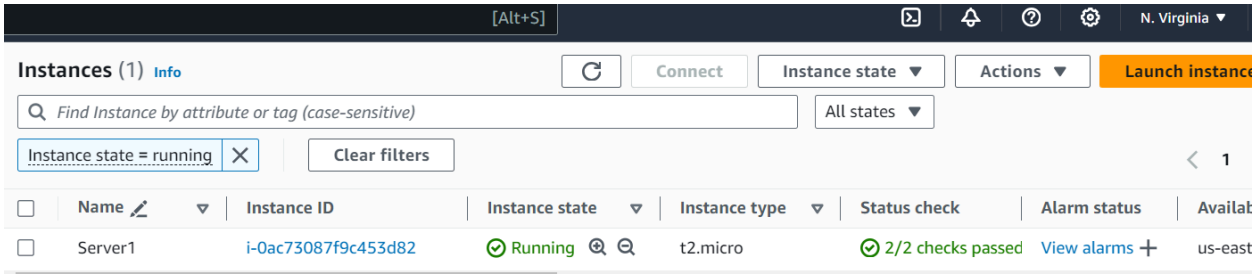
Launch Linux EC2 instances in two regions using a single Terraform file.

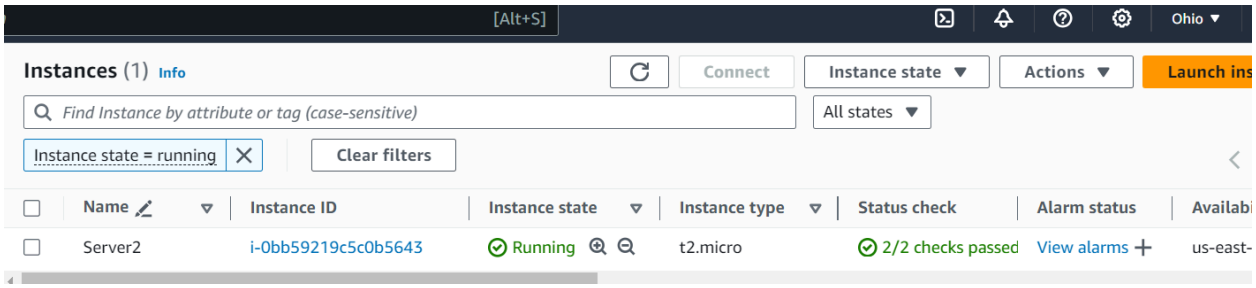
```
ubuntu@ip-172-31-44-140:~$ aws --version
aws-cli/1.22.34 Python/3.10.12 Linux/6.5.0-1017-aws botocore/1.23.34
ubuntu@ip-172-31-44-140:~$ terraform --version
Terraform v1.8.2
on linux_amd64
ubuntu@ip-172-31-44-140:~$ mkdir server
ubuntu@ip-172-31-44-140:~$ ls
server
ubuntu@ip-172-31-44-140:~$ cd server/
ubuntu@ip-172-31-44-140:~/server$ vi file.tf
```

```
ubuntu@ip-172-31-44-140:~/server$ cat file.tf
provider "aws" {
     alias  = "us-east-1"
     region = "us-east-1"
}

provider "aws" {
     alias  = "us-east-2"
     region = "us-east-2"
}

resource "aws_instance" "launch" {
       ami           = "ami-04b70fa74e45c3917"
       instance_type = "t2.micro"
       provider      = aws.us-east-1
       tags = {
            Name     = "Server1"
       }
}

resource "aws_instance" "launch1" {
       ami           = "ami-09040d770ffe2224f"
       instance_type = "t2.micro"
       provider      = aws.us-east-2
       tags = {
            Name     = "Server2"
       }
}
ubuntu@ip-172-31-44-140:~/server$
```

```
ubuntu@ip-172-31-44-140:~/server$ terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.48.0

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-172-31-44-140:~/server$ 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.launch will be created
  + resource "aws_instance" "launch" {
      + ami                                  = "ami-04b70fa74e45c3917"
      + 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)
      + 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" = "Server2"
        }
      + tags_all                             = {
          + "Name" = "Server2"
        }
      + 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: 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-172-31-44-140:~/server$ 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.launch will be created
```

```
      + 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" = "Server2"
        }
      + tags_all                             = {
          + "Name" = "Server2"
        }
      + 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: 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.launch: Creating...
aws_instance.launch1: Creating...
aws_instance.launch: Still creating... [10s elapsed]
aws_instance.launch1: Still creating... [10s elapsed]
aws_instance.launch: Still creating... [20s elapsed]
aws_instance.launch1: Still creating... [20s elapsed]
aws_instance.launch: Still creating... [30s elapsed]
aws_instance.launch1: Still creating... [30s elapsed]
aws_instance.launch1: Creation complete after 35s [id=i-0bb59219c5c0b5643]
aws_instance.launch: Creation complete after 36s [id=i-0ac73087f9c453d82]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```


[Alt+S]
N. Virginia
Instances (1) Info
Connect
Instance state
Actions
Launch instance
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availab
Server1
i-0ac73087f9c453d82
Running
t2.micro
2/2 checks passed
View alarms
us-east


[Alt+S]
Ohio
Instances (1) Info
Connect
Instance state
Actions
Launch ins
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availabi
Server2
i-0bb59219c5c0b5643
Running
t2.micro
2/2 checks passed
View alarms
us-east-