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

```
PS C:\Users\Microsense> mkdir terraformtasknew
    Directory: C:\Users\Microsense


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         3/19/2025  12.30 PM                terraformtasknew
```

Create one directory

```
Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         3/19/2025  12.30 PM                terraformtasknew


PS C:\Users\Microsense> cd  terraformtasknew
PS C:\Users\Microsense\terraformtasknew> code .
```

Then cd INTO THAT DIRECTORY

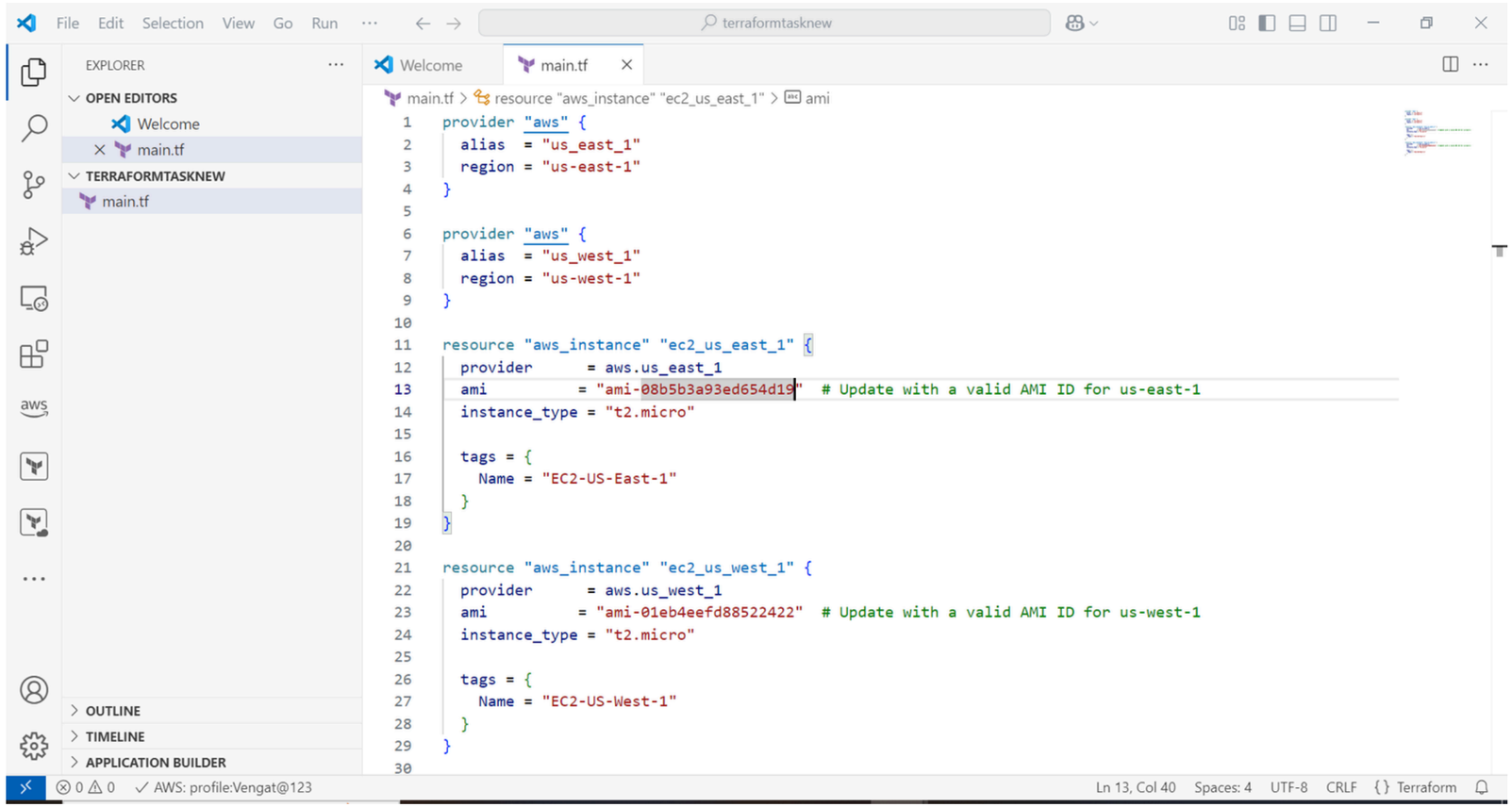
- Before that
- Install Terraform
- Configure AWS Cli

```
PS C:\Users\Microsense\terraformtasknew> terraform --version
Terraform v1.9.8
on windows_amd64

Your version of Terraform is out of date! The latest version
is 1.11.2. You can update by downloading from https://www.terraform.io/downloads.html
PS C:\Users\Microsense\terraformtasknew>
```

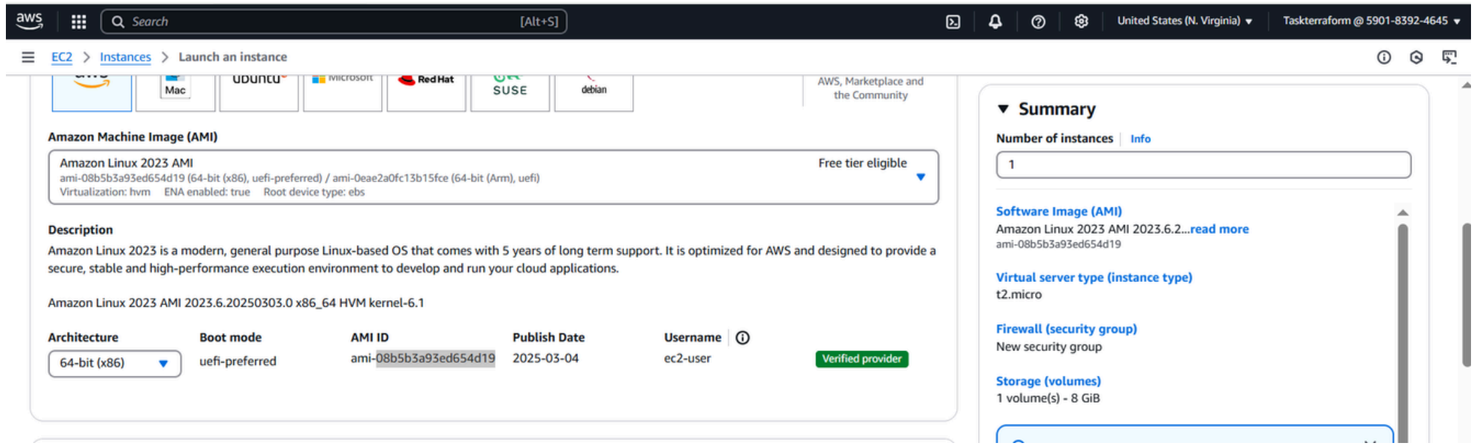
Terraform Version

```
PS C:\Users\Microsense\terraformtasknew> aws configure
AWS Access Key ID [****************OJ5T]: AKIAYS2NUAOS7EHBOJ5T
AWS Secret Access Key [****************t2YD]: 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
Default region name [None]:
Default output format [None]:
PS C:\Users\Microsense\terraformtasknew>
```

Aws Configure

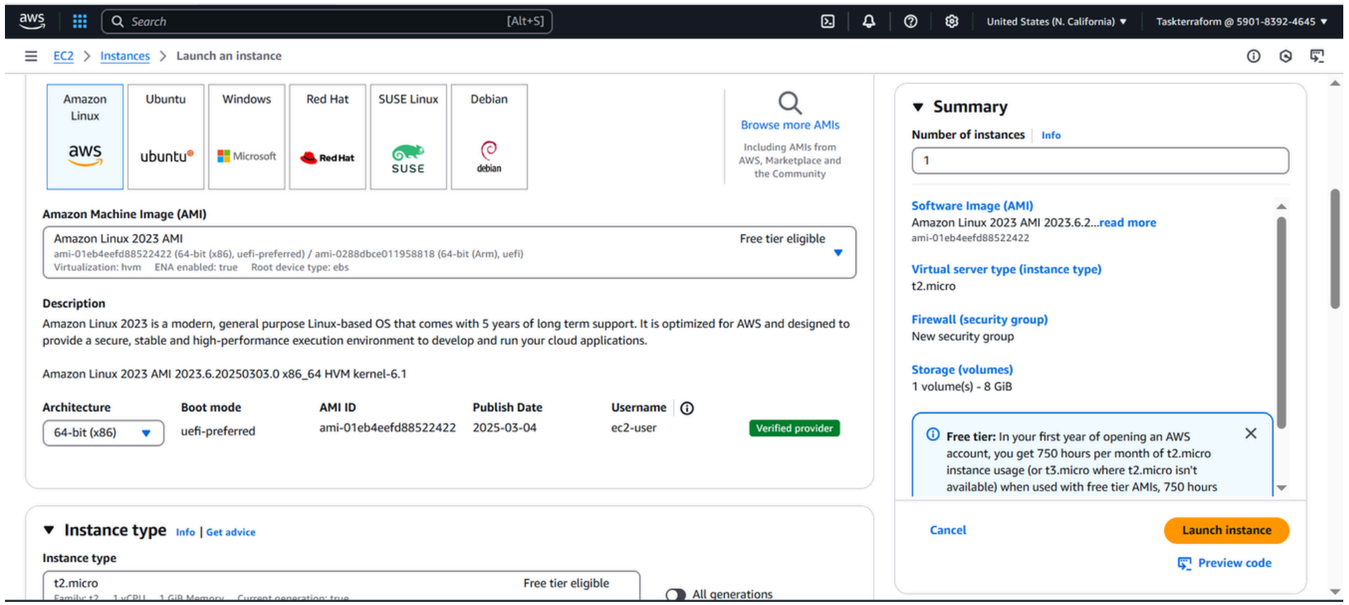


Create Main.tf File

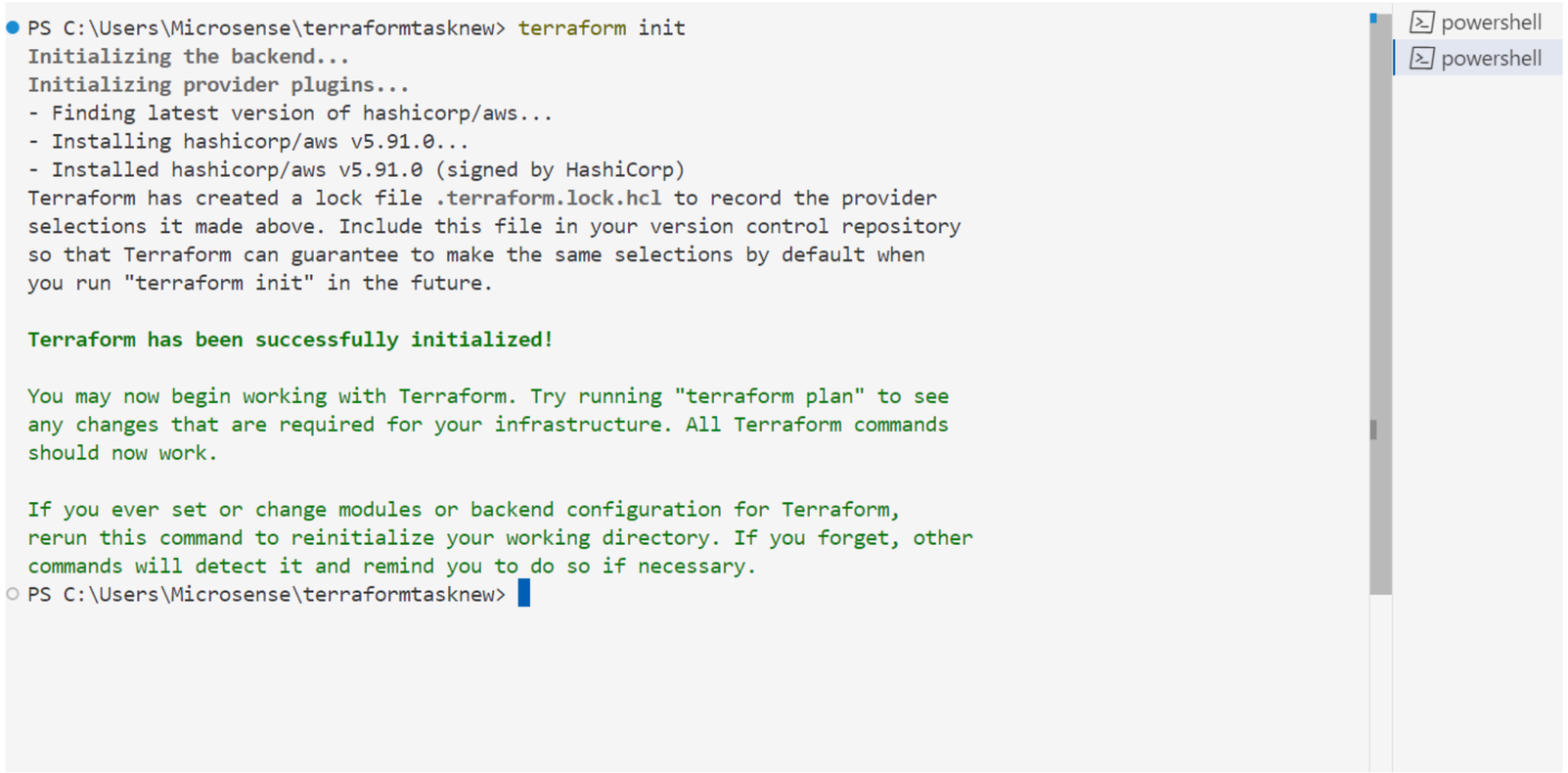


us-east-1 ami

Copy the instance ami and past in vs code ami



us-west-1 ami



Terraform Init

Terraform plan

```
Terraform will perform the following actions:

  # aws_instance.ec2_us_east_1 will be created
  + resource "aws_instance" "ec2_us_east_1" {
      + ami                                  = "ami-08b5b3a93ed654d19"
      + 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)
      + 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)
```

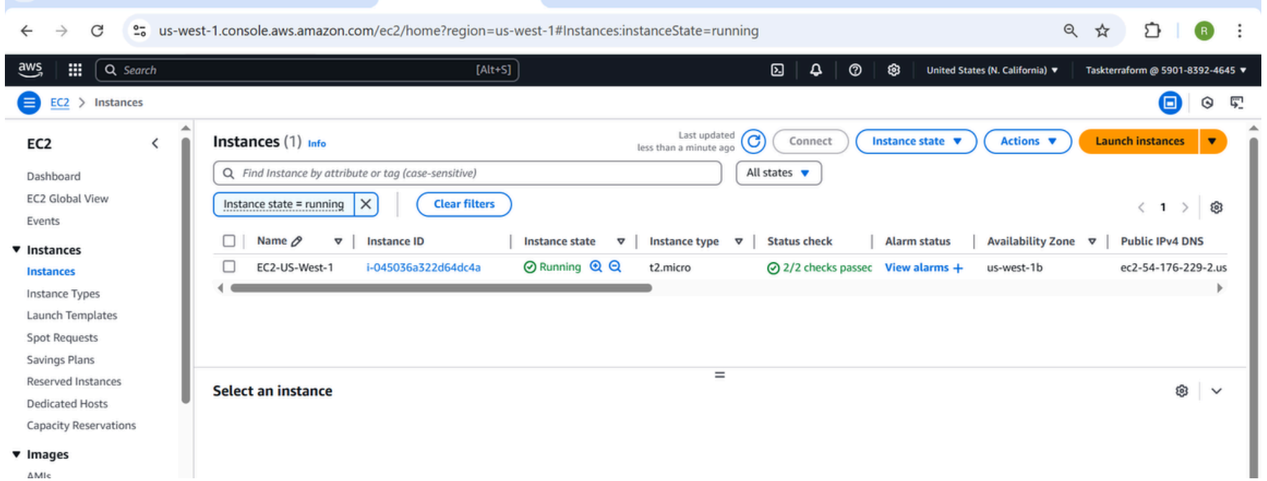
```
    }

  # aws_instance.ec2_us_west_1 will be created
  + resource "aws_instance" "ec2_us_west_1" {
      + ami                                  = "ami-01eb4eefd88522422"
      + 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)
      + 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)
```

```
      + 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.
PS C:\Users\Microsense\terraformtasknew>
```

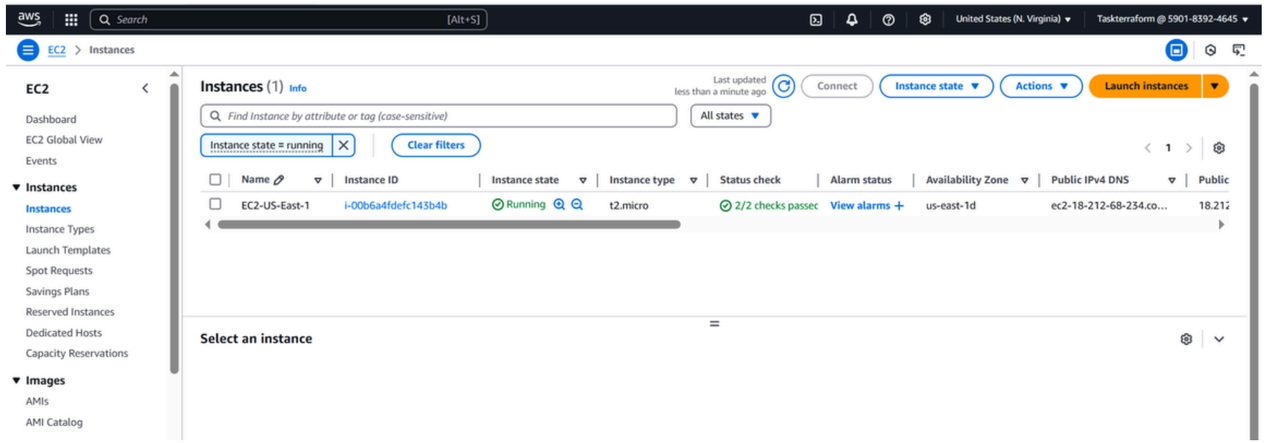
Terraform apply

```
PS C:\Users\Microsense\terraformtasknew> 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.ec2_us_east_1 will be created
  + resource "aws_instance" "ec2_us_east_1" {
      + ami                                  = "ami-08b5b3a93ed654d19"
      + 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)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + 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.ec2_us_west_1: Creating...
aws_instance.ec2_us_east_1: Creating...
aws_instance.ec2_us_east_1: Still creating... [10s elapsed]
aws_instance.ec2_us_west_1: Still creating... [10s elapsed]
aws_instance.ec2_us_west_1: Creation complete after 19s [id=i-045036a322d64dc4a]
aws_instance.ec2_us_east_1: Creation complete after 19s [id=i-00b6a4fdefc143b4b]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\Microsense\terraformtasknew>
```

powershe
powershe



Ec2 instance created in us-west-1



Ec2 instance created in us-east-1