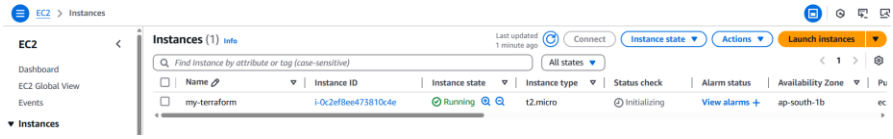
Terraform Assignment - 1

You have been asked to:

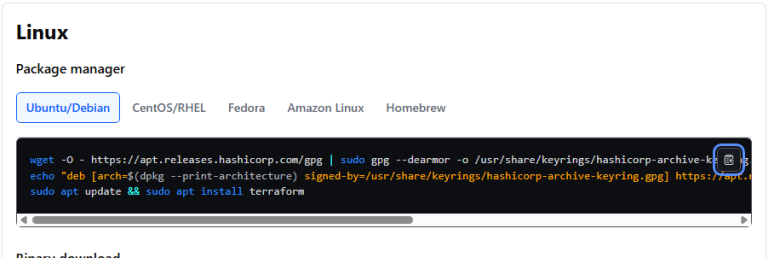
- Create an EC2 service in the default subnet in the ohio region

Solution:

1) Launch EC2 instance and download terraform from official website :



2) Connect and download terraform :



3) Verify the Installation: terraform version

```
ubuntu@ip-172-31-7-35:~$ terraform version
Terraform v1.11.2
on linux_amd64
ubuntu@ip-172-31-7-35:~$
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

4) Create a new directory for your Terraform project:
mkdir terraform-project
cd terraform-project

```
ubuntu@ip-172-31-7-35:~$ mkdir terraform-project
ubuntu@ip-172-31-7-35:~$ cd terraform-project
ubuntu@ip-172-31-7-35:~/terraform-project$
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

5) Create a new file named main.tf for your Terraform configuration:
touch main.tf
nano main.tf
6) Add your Terraform code to main.tf:

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

resource "aws_instance" "example" {
  ami           = "ami-0cb91c7de36eed2cb"
  instance_type = "t2.micro"
}
```

```
^G Help          ^O Write Out     ^W Where Is
^X Exit          ^R Read File     ^\ Replace
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

7) Terraform needs AWS credentials to interact with AWS services:
Attach an IAM role to your EC2 instance with sufficient permissions > **EC2 -> Instances -> Actions -> Security -> Modify IAM Role** > choose instance profile and ec2 full access role [create if there is no role]

8) Initialize Terraform: terraform init

```
ubuntu@ip-172-31-7-35:~/terraform-project$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.91.0...
- Installed hashicorp/aws v5.91.0 (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-172-31-7-35:~/terraform-project$
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

9) Plan and Apply the Configuration:
terraform plan

```
ubuntu@ip-172-31-7-35:~/terraform-project$ terraform plan

Terraform used the selected providers to generate the following executio
  + create

Terraform will perform the following actions:

  # aws_instance.example will be created
  + resource "aws_instance" "example" {
      + ami                                  = "ami-0cb91c7de36eed2cb"
      + 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)
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

terraform apply

```
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.example: Creating...
aws_instance.example: Still creating... [10s elapsed]
aws_instance.example: Creation complete after 15s [id=i-0f77ee2e98c7ae98f]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-7-35:~/terraform-project$
```

i-0c2ef8ee473810c4e (my-terraform)

PublicIPs: 3.110.54.55 PrivateIPs: 172.31.7.35

10) Verify : go to ohio region and we can see the EC2 we created.