Terraform Assignment - 1

You have been asked to:
Create an EC2 service in the default subnet in the Ohio region

Answer:

```
$ sudo apt-get update

$ wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg

$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-
keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs)
main" | sudo tee /etc/apt/sources.list.d/hashicorp.list

$ sudo apt update && sudo apt install terraform
$ mkdir assignment1

$ cat var.tf

variable "access" {
        type = string
        default = "SHYTGGKEI"
}

variable "secret" {
        type = string
        default ="+LS6N0C6oiKtHmlUh5o"
}

$ cat main.tf

provider "aws" {
        region = "us-east-2"
        secret_key = var.secret
        access_key = var.access
}
resource "aws_instance" "Ins1" {
        ami = "ami-05fb0b8c1424f266b"
        instance_type = "t2.micro"
        key_name = "keypair_ravi"
        tags = {
                Name = "Assignment-1"
        }
}
```

```
ubuntu@ip-172-31-10-221:~/assignment1$ sudo terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.33.0...
- Installed hashicorp/aws v5.33.0 (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.
```

```
sudo terraform plan
sudo terraform apply
```

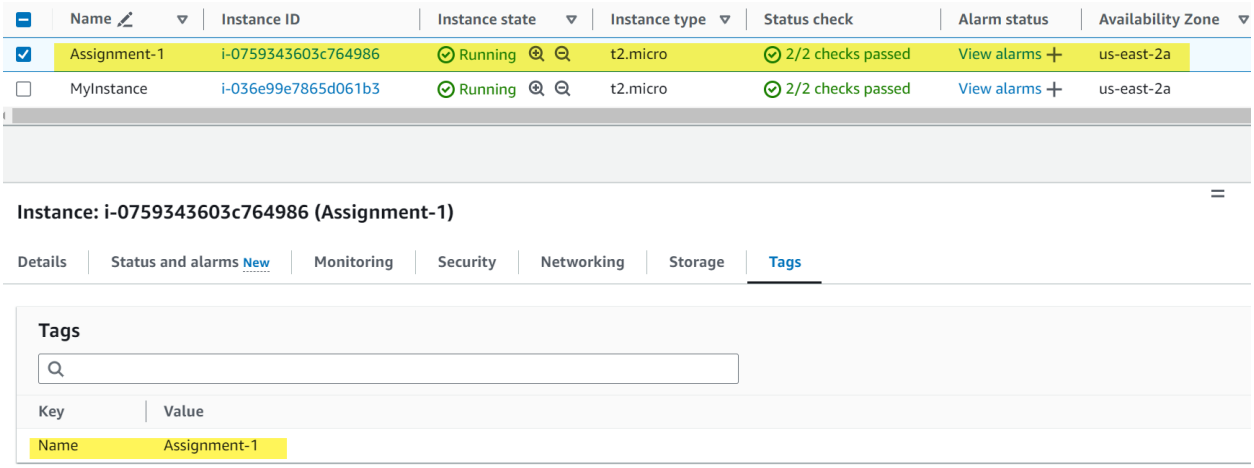
```
Plan: 1 to add, 0 to change, 0 to destroy.

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.Ins1: Creating...
aws_instance.Ins1: Still creating... [10s elapsed]
aws_instance.Ins1: Still creating... [20s elapsed]
aws_instance.Ins1: Still creating... [30s elapsed]
aws_instance.Ins1: Creation complete after 32s [id=i-0759343603c764986]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-10-221:~/assignment1$
```



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Assignment-1	i-0759343603c764986	Running	t2.micro	2/2 checks passed	View alarms	us-east-2a
MyInstance	i-036e99e7865d061b3	Running	t2.micro	2/2 checks passed	View alarms	us-east-2a

Instance: i-0759343603c764986 (Assignment-1)

Details | Status and alarms New | Monitoring | Security | Networking | Storage | Tags

Tags

Key	Value
Name	Assignment-1