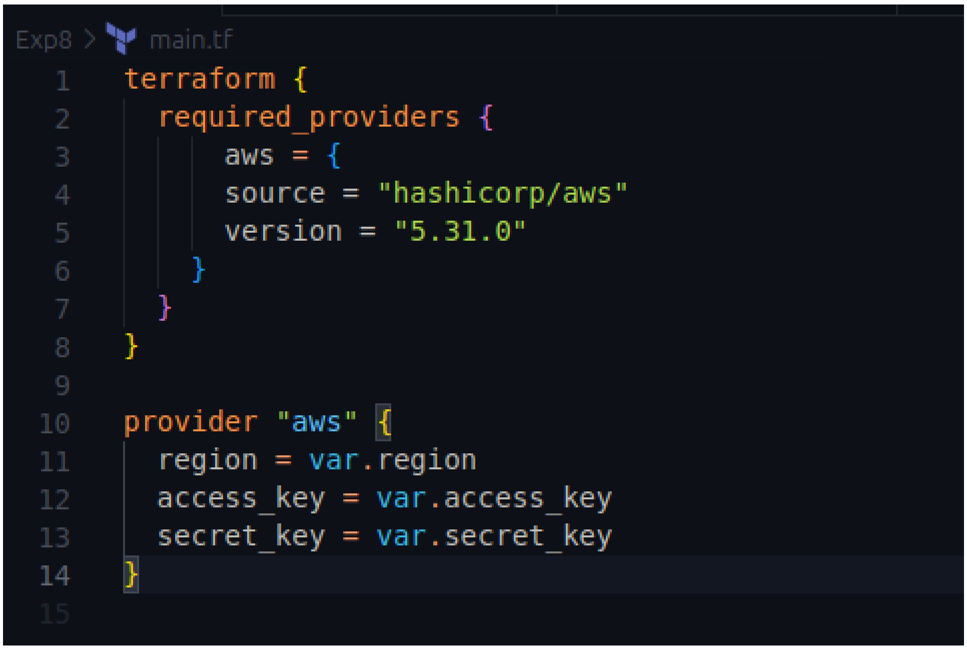
Experiment 9

Creating multiple EC2 Instances with for_each in Terraform

Steps

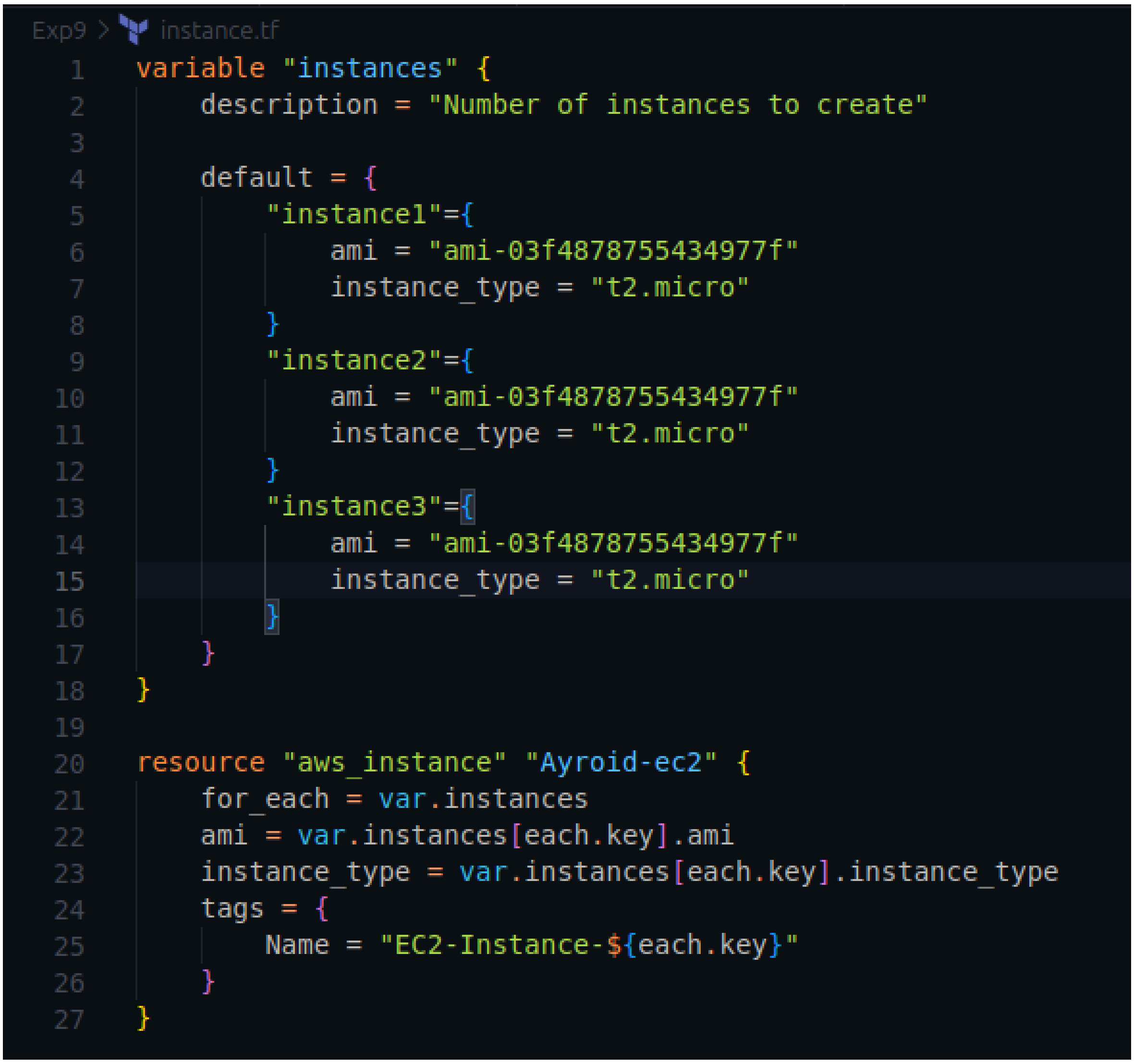
1. Create a main.tf file



Exp8 > main.tf

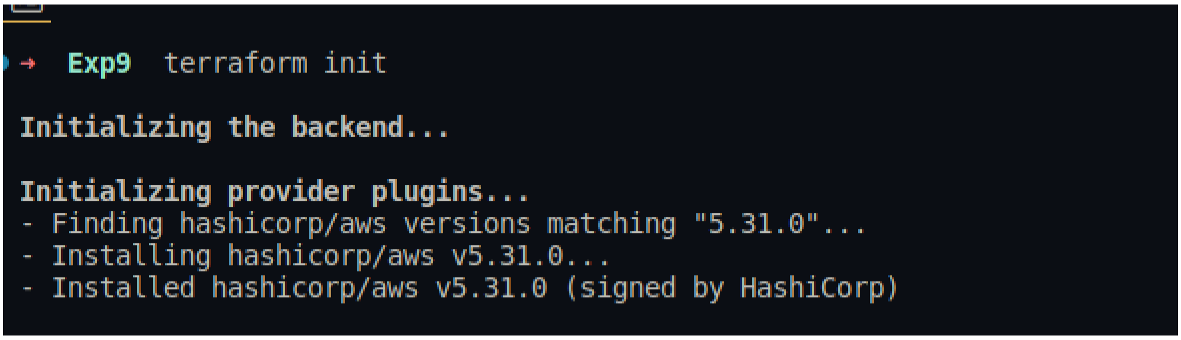
```
terraform {
  required_providers {
    aws = {
    source = "hashicorp/aws"
    version = "5.31.0"
    }
  }
}

provider "aws" {
  region = var.region
  access_key = var.access_key
  secret_key = var.secret_key
}
```

2. Create instance.tf file

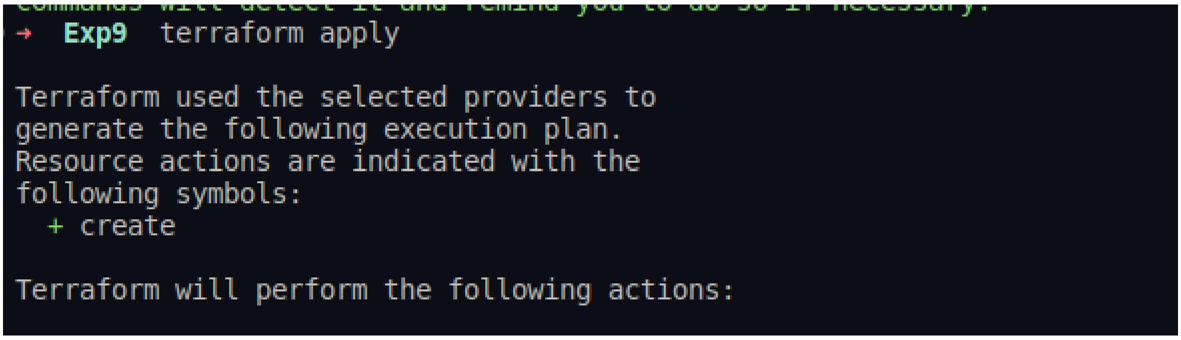


```
variable "instances" {
    description = "Number of instances to create"

    default = {
        "instance1"={
            ami = "ami-03f4878755434977f"
            instance_type = "t2.micro"
        }
        "instance2"={
            ami = "ami-03f4878755434977f"
            instance_type = "t2.micro"
        }
        "instance3"={
            ami = "ami-03f4878755434977f"
            instance_type = "t2.micro"
        }
    }
}

resource "aws_instance" "Ayroid-ec2" {
    for_each = var.instances
    ami = var.instances[each.key].ami
    instance_type = var.instances[each.key].instance_type
    tags = {
        Name = "EC2-Instance-${each.key}"
    }
}
```

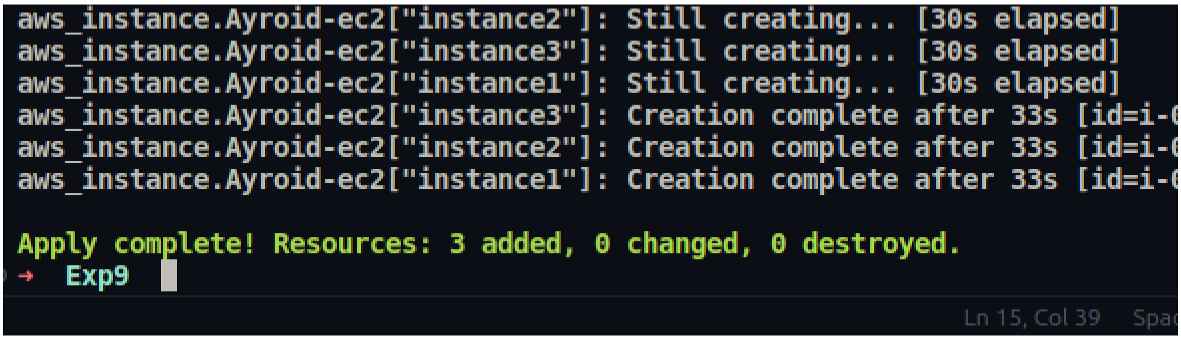
3. Run Terraform init and apply commands



```
➜  Exp9  terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (signed by HashiCorp)
```

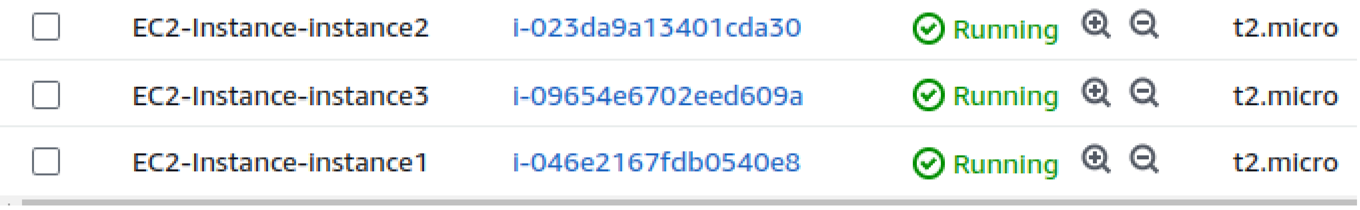


```
commands will detect it and remind you to do so if necessary.
➜  Exp9  terraform apply

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:
```



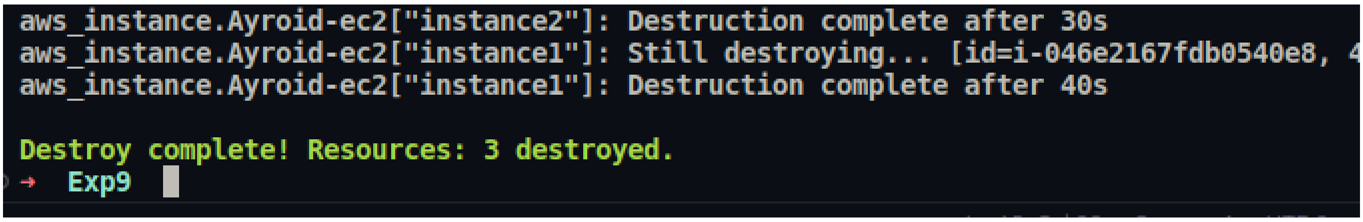
```
aws_instance.Ayroid-ec2["instance2"]: Still creating... [30s elapsed]
aws_instance.Ayroid-ec2["instance3"]: Still creating... [30s elapsed]
aws_instance.Ayroid-ec2["instance1"]: Still creating... [30s elapsed]
aws_instance.Ayroid-ec2["instance3"]: Creation complete after 33s [id=i-0
aws_instance.Ayroid-ec2["instance2"]: Creation complete after 33s [id=i-0
aws_instance.Ayroid-ec2["instance1"]: Creation complete after 33s [id=i-0

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
➜  Exp9
```

4. Verify resources on AWS



☐	EC2-Instance-instance2	i-023da9a13401cda30	✓ Running	t2.micro
☐	EC2-Instance-instance3	i-09654e6702eed609a	✓ Running	t2.micro
☐	EC2-Instance-instance1	i-046e2167fdb0540e8	✓ Running	t2.micro

5. Clean up resources



```
aws_instance.Ayroid-ec2["instance2"]: Destruction complete after 30s
aws_instance.Ayroid-ec2["instance1"]: Still destroying... [id=i-046e2167fdb0540e8, 4
aws_instance.Ayroid-ec2["instance1"]: Destruction complete after 40s

Destroy complete! Resources: 3 destroyed.
➜  Exp9
```