

Experiment 4

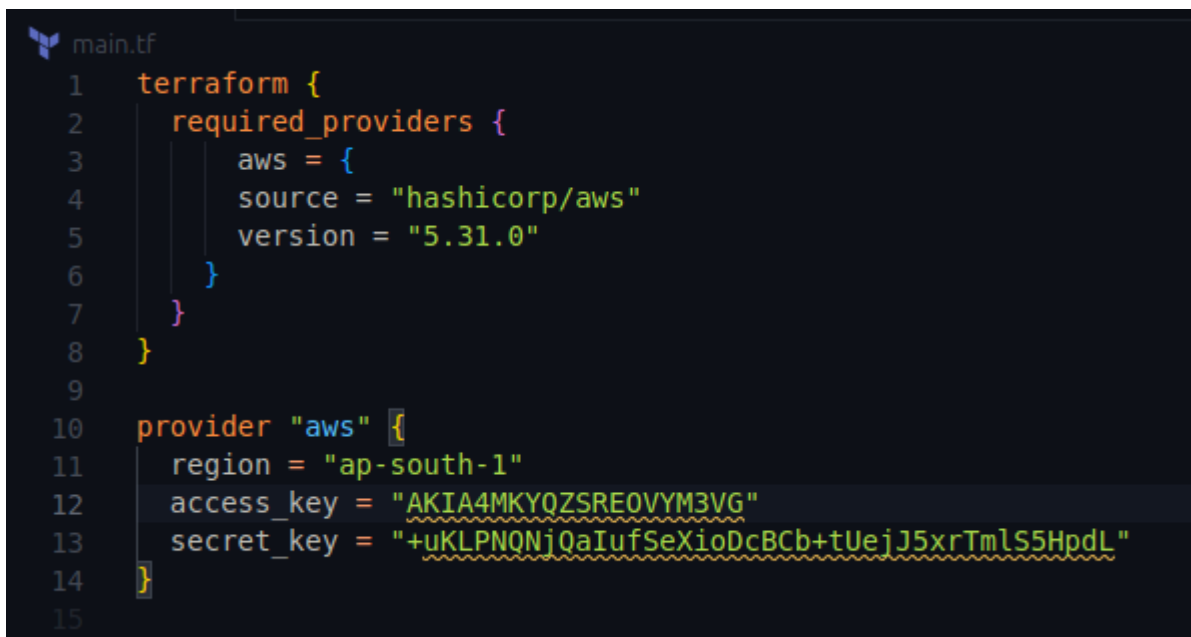
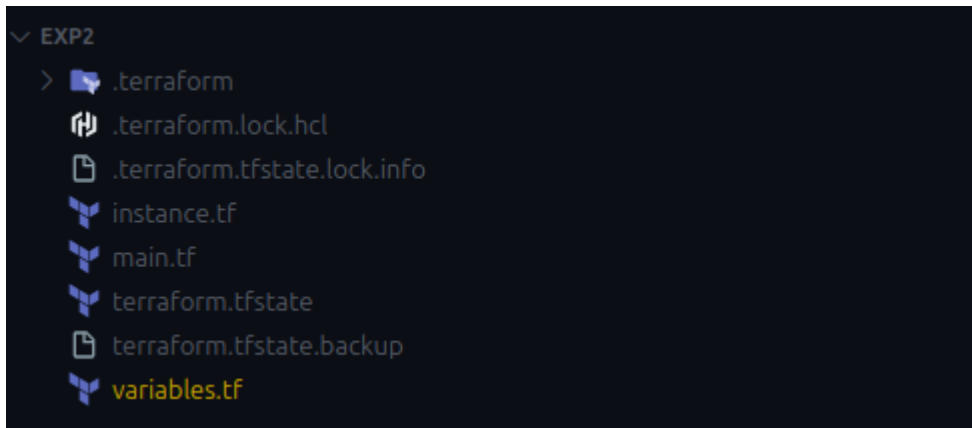
Terraform Variables

Aim

Learn how to define and use variables in Terraform configuration

Steps

1. Create a main file & terraform configuration file for EC2 Instance (instance.tf)



```

instance.tf
1  resource "aws_instance" "Ayroid-ec2" {
2      instance_type = "t2.micro"
3      ami = "ami-03f4878755434977f"
4      count = 1
5
6      tags = {
7          Name = "Exp4-Instance"
8      }
9  }

```

2. Open a new file named variables.tf. Define variables for region, ami, secret_key, access_key and instance_type.

```

variables.tf
1  variable region {
2      type      = string
3      default   = "ap-south-1"
4      description = "AWS Region"
5  }
6
7  variable "ami"{
8      type = string
9      default = "ami-03f4878755434977f"
10     description = "AMI ID"
11 }
12
13 variable "instance_type"{
14     type = string
15     default = "t2.micro"
16     description = "Instance Type"
17 }
18

```

3. Modify main.tf & instance.tf to use the variables.

```

instance.tf x
instance.tf
1  resource "aws_instance" "Ayroid-ec2" {
2      instance_type = var.instance_type
3      ami = var.ami
4      count = 1
5
6      tags = {
7          Name = "Exp4-Instance"
8      }
9  }

```

```

main.tf
1  terraform {
2      required_providers {
3          aws = {
4              source = "hashicorp/aws"
5              version = "5.31.0"
6          }
7      }
8  }
9
10 provider "aws" {
11     region = var.region
12     access_key = var.access_key
13     secret_key = var.secret_key
14 }
15

```

4. Run the following Terraform commands to initialize and apply the configuration.

```

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.Ayroid-ec2[0]: Creating...
aws_instance.Ayroid-ec2[0]: Still creating... [10s elapsed]
aws_instance.Ayroid-ec2[0]: Still creating... [20s elapsed]
aws_instance.Ayroid-ec2[0]: Still creating... [30s elapsed]
aws_instance.Ayroid-ec2[0]: Creation complete after 34s [id=i-0688471057f5dbedd]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```

5. Verify Resources

Instances (1) Info				
<input type="text"/> Find Instance by attribute or tag (case-sensitive)				Any
<div> <div>Instance state = running</div> <div>×</div> </div>		<div>Clear filters</div>		
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type
<input type="checkbox"/>	Exp4-Instance	I-04d94f7cc982883d5	Running	t2.micro

6. Cleanup Resources

```
• → Exp2 terraform destroy
aws_instance.Ayroid-ec2[0]: Refreshing state... [id=i-0688471057f5dbedd]

Terraform used the selected
providers to generate the
following execution plan.
Resource actions are indicated
with the following symbols:
- destroy

Terraform will perform the following actions:

# aws_instance.Ayroid-ec2[0] will be destroyed
- resource "aws_instance" "Ayroid-ec2" {
  - ami              = "ami-03f4878755434977f" -> null
  - arn              = "arn:aws:ec2:ap-south-1:851931352354:instance/i-068847
```