Terraform Concepts & Working

Make some folders and place your code.

$ mkdir terraform-demo
$ cd terraform-demo
$ mkdir demo1
$ cd demo1

Create a tf file within demo1
$ vim example1.tf

For AMI visit link - https://aws.amazon.com/amazon-linux-ami/

Example 1 - First EC2 instance

```
provider "aws" {
 region = "ap-south-1"
 access_key = "<Access-Key>"
 secret_key = "<Secret-Key>"
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
}
```

// Save the above content in file and follow below commands

$ terraform init
$ terraform plan
$ terraform apply

Modify existing file and write below example -

Example 2 - AWS Authentication using shared credentials file
(NOTE: Use aws configure command before to add the AWS credentials to .aws/credentials file)

```
provider "aws" {
region = "ap-south-1"
}
resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
```

```
 tags = {
    "Name"        = "app-server"
    "environment" = "dev"
  }
}
resource "aws_instance" "instance02" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "appserver"
    "environment" = "stage"
  }
 }
```

$ terraform plan
$ terraform apply

Example 3 - Change in the infrastructure

```
provider "aws" {
region = "ap-south-1"
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = "dev"
  }
}
resource "aws_eip" "newIP" {
 instance = aws_instance.instance01.id
 vpc = true
}
```

$ terraform plan
$ terraform apply

How to write comment in terraform

```
# Single line comment

//Single line comment

/*
Block comment
*/
```

Example 4 - View the tfstate file content in terraform

$ terraform show

Example 5 - Destroy the infrastructure

$ terraform destroy

Example 6 - Resource Dependency // Implicit & Explicit

```
provider "aws" {
 region = "ap-south-1"
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = "dev"
 }
depends_on = [aws_ebs_volume.diskSize]
}

resource "aws_ebs_volume" "diskSize" {
 availability_zone = "ap-south-1a"
 size = 10
}

resource "aws_volume_attachment" "ebs_add" {
  device_name = "/dev/xvdf"
  volume_id   = aws_ebs_volume.diskSize.id
  instance_id = aws_instance.instance01.id
}

resource "aws_eip" "newIP" {
 instance = aws_instance.instance01.id
 vpc = true
}
```

$ terraform apply

Example 7 - Provision local/external

```
provider "aws" {
 region = "ap-south-1"
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = "dev"
 }
  provisioner "local-exec" {
    command = "echo ${aws_instance.instance01.public_ip} > ip_address.txt"
 }

}
```

$ terraform apply

```
provider "aws" {
 region = "ap-south-1"
}

resource "aws_instance" "instance01" {
 ami = "ami-09ba48996007c8b50"
 instance_type = "t2.micro"
 tags = {
 "Name" = "terraform-22"
 "environment" = "dev"
 }
   provisioner "remote-exec" {
    inline = [
      "sudo amazon-linux-extras enable nginx1.12",
      "sudo yum -y install nginx",
      "sudo systemctl start nginx",
    ]
  }
}
```

$ terraform apply

Example 8 - Defining Variable - Input / Output Variable

```
variable "region" {
  default = "ap-south-1"
```

```
}

provider "aws" {
 region = var.region
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = "dev"
  }
}

output "ip" {
  value = aws_instance.instance01.public_ip

}
```

Example 9 - Defining Local Variable

```
locals {
 env = "forum"
 owner = "Ritesh-DevOps"
}

variable "region" {
  default = "ap-south-1"
}

provider "aws" {
 region = var.region
}

resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = local.env
    "owner" = local.owner
  }
}

output "ip" {
  value = aws_instance.instance01.public_ip
```

```
}
```

#Example 10 - Splitting the input, output and provider in different files

---

vim example.tf

```
resource "aws_instance" "instance01" {
 ami = "ami-04db49c0fb2215364"
 instance_type = "t2.micro"
 tags = {
    "Name"        = "web-server"
    "environment" = "dev"
 }
}
```

---

vim variables.tf

```
variable "region" {
  default = "ap-south-1"
}
```

---

vim outputs.tf

```
output "ip" {
  value = aws_instance.instance01.public_ip
}
```

---

vim provider.tf

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

Name: Ritesh Goyal
Contact: 9960930111
Email-id: ritesh.devopstrainer@gmail.com