Detailed Terraform Examples

1. **Backend configuration**
   Each Terraform configuration can specify a backend, which defines where state snapshots are stored.

```
terraform {
  backend "s3" {
    bucket = "aws-infrastructure-tfstate"
    key    = "devops/terraform.tfstate"
    region = "ap-south-1"
  }
}
```

2. **Example - SSH to server through security group**

```
// generate ssh-key pair from AWS console

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

resource "aws_instance" "instance01" {
 ami = "ami-09ba48996007c8b50"
 instance_type = "t2.micro"
 key_name = "ritesh-devops-demo"
 tags = {
    Name = "terraform-devops"
  }
 security_groups = ["${aws_security_group.AWSaccess.name}"]
}

resource "aws_security_group" "AWSaccess" {
 name = "AWSaccess"
 description = "SSH access"

 ingress {
 protocol = "tcp"
 from_port = 22
 to_port = 22
 cidr_blocks = ["0.0.0.0/0"]
 }
}
```

3. **Remote command execution using provisioner**

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

```
}

resource "aws_instance" "instance01" {
 ami = "ami-09ba48996007c8b50"
 instance_type = "t2.micro"
 key_name =  "ritesh-devops-demo"
 tags = {
 "Name" = "web-server"
 "environment" = "dev"
 }
  provisioner "remote-exec" {
    inline = [
      "sudo amazon-linux-extras install -y nginx1",
      "sudo systemctl start nginx"
    ]
    connection {
          type     = "ssh"
          user     = "ec2-user"
          private_key = file("../devops-ritesh.pem")
          host     = self.public_ip
          timeout = "30m"
    }
  }

  security_groups = ["${aws_security_group.AWSaccess.name}"]
}

resource "aws_security_group" "AWSaccess" {
  name        = "AWSaccess-dev"
  description = "SSH access"

  ingress {
    protocol    = "tcp"
    from_port   = 22
    to_port     = 22
    cidr_blocks = ["0.0.0.0/0"]
  }
    ingress {
    protocol    = "tcp"
    from_port   = 80
    to_port     = 80
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
```

```
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

4. Example - Custom key-pair value

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

resource "aws_key_pair" "awsKey" {
 key_name = "devops-serverKey"
 public_key = "${file("devops-serverKey.pub")}"
}

resource "aws_instance" "instance01" {
 ami = "ami-09ba48996007c8b50"
 instance_type = "t2.micro"
 key_name = aws_key_pair.awsKey.key_name
 tags = {
    Name = "terraform-example"
  }
 security_groups = ["${aws_security_group.AWSaccess.name}"]
}

resource "aws_security_group" "AWSaccess" {
 name = "AWSaccess"
 description = "SSH access"

 ingress {
 protocol = "tcp"
 from_port = 22
 to_port = 22
 cidr_blocks = ["0.0.0.0/0"]
 }
}
```

5. Example - Search ami and use it in resource

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

data "aws_ami" "amazon_linux_2" {
  most_recent = true
  owners      = ["amazon"]
```

```
  filter {
    name   = "name"
    values = ["amzn2-ami-hvm*"]
  }
}

output "amazon-ami" {
  value = "${data.aws_ami.amazon_linux_2.id}"
}

resource "aws_key_pair" "awsKey" {
 key_name = "appserverKey"
 public_key = "${file("$HOME/.ssh/id_rsa.pub")}"
}

resource "aws_instance" "instance01" {
 ami = data.aws_ami.amazon_linux_2.id
 instance_type = "t2.micro"
 key_name = aws_key_pair.awsKey.key_name
 tags = {
    Name = "terraform-example"
  }
 security_groups = ["${aws_security_group.AWSaccess.name}"]
}

resource "aws_security_group" "AWSaccess" {
 name = "AWSaccess"
 description = "SSH access"

 ingress {
 protocol = "tcp"
 from_port = 22
 to_port = 22
 cidr_blocks = ["0.0.0.0/0"]
 }
}
```

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