

# Terraform Assignment - 4

You have been asked to:

- Destroy the previous deployments

```
Plan: 0 to add, 0 to change, 2 to destroy.

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_instance.assignment-3-1: Destroying... [id=i-0cea61bac395eef32]
aws_instance.assignment-3-2: Destroying... [id=i-0cd013ea250a502be]
aws_instance.assignment-3-1: Still destroying... [id=i-0cea61bac395eef32, 10s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-0cd013ea250a502be, 10s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-0cea61bac395eef32, 20s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-0cd013ea250a502be, 20s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-0cea61bac395eef32, 30s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-0cd013ea250a502be, 30s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-0cea61bac395eef32, 40s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-0cd013ea250a502be, 40s elapsed]
aws_instance.assignment-3-2: Destruction complete after 40s
aws_instance.assignment-3-1: Still destroying... [id=i-0cea61bac395eef32, 50s elapsed]
aws_instance.assignment-3-1: Destruction complete after 50s

Destroy complete! Resources: 2 destroyed.
root@ip-10-0-0-195:~# terraform apply
```

- Create a VPC with the required components using Terraform
- Deploy an EC2 instance inside the VPC

```
Enter a value: yes

aws_route_table_association.example_rta: Destroying... [id=rtbassoc-0aaaca8f472b2e7bb]
aws_route_table_association.example_rta: Destruction complete after 0s
aws_subnet.example_subnet: Destroying... [id=subnet-03138947f5f5a12a5]
aws_subnet.example_subnet: Destruction complete after 1s
aws_subnet.example_subnet: Creating...
aws_subnet.example_subnet: Creation complete after 0s [id=subnet-08e8ef4c1e1258b26]
aws_instance.example_instance: Creating...
aws_route_table_association.example_rta: Creating...
aws_route_table_association.example_rta: Creation complete after 1s [id=rtbassoc-0d7ba68951e106e91]
aws_instance.example_instance: Still creating... [10s elapsed]
aws_instance.example_instance: Still creating... [20s elapsed]
aws_instance.example_instance: Still creating... [30s elapsed]
aws_instance.example_instance: Creation complete after 33s [id=i-0aa814d0272645821]

Apply complete! Resources: 3 added, 0 changed, 2 destroyed.
root@ip-172-31-14-245:~#
```

Code:

```
provider "aws" {
  region = "us-west-2"
  access_key = "AKIAVNPLXOGUAGQNP6LT"
  secret_key = "5J1CWJvqwFQh50sT5u789ndFvmWc+X2bllf96neJ"
```

```

}

resource "aws_vpc" "example_vpc" {
  cidr_block = "10.0.0.0/16"
  enable_dns_support = true
  enable_dns_hostnames = true

  tags = {
    Name = "example-vpc"
  }
}

resource "aws_subnet" "example_subnet" {
  vpc_id = aws_vpc.example_vpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-west-2a"
  tags = {
    Name = "example-subnet"
  }
}

resource "aws_internet_gateway" "example_igw" {
  vpc_id = aws_vpc.example_vpc.id

  tags = {
    Name = "example-igw"
  }
}

resource "aws_route_table" "example_rt" {
  vpc_id = aws_vpc.example_vpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.example_igw.id
  }

  tags = {
    Name = "example-rt"
  }
}

resource "aws_route_table_association" "example_rta" {
  subnet_id = aws_subnet.example_subnet.id

```

```

    route_table_id = aws_route_table.example_rt.id
  }

resource "aws_security_group" "example_sg" {
  name      = "example-sg"
  description = "Allow SSH inbound traffic"
  vpc_id    = aws_vpc.example_vpc.id

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

  egress {
    from_port = 0
    to_port   = 0
    protocol  = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "example-sg"
  }
}

resource "aws_instance" "example_instance" {
  ami          = "ami-08f7912c15ca96832"
  instance_type = "t2.micro"
  subnet_id    = aws_subnet.example_subnet.id
  vpc_security_group_ids = [aws_security_group.example_sg.id]
  tags = {
    Name = "example-instance"
  }
}

```

Code 2: Shorter

```

provider "aws" {
  region = "us-west-2"
  access_key = "AKIAVNPLXOGUAGQNP6LT"
  secret_key = "5J1CWJvqwFQh50sT5u789ndFvmWc+X2bllf96neJ"
}

```

```

resource "aws_vpc" "example_vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name = "example-vpc"
  }
}
resource "aws_subnet" "example_subnet" {
  vpc_id     = aws_vpc.example_vpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-west-2a"
  tags = {
    Name = "example-subnet"
  }
}

resource "aws_instance" "example_instance" {
  ami          = "ami-08f7912c15ca96832"
  instance_type = "t2.micro"
  key_name     = "WOW"
  subnet_id    = aws_subnet.example_subnet.id
  tags = {
    Name = "example-instance"
  }
}

```

Code 3: more shorter

```

provider "aws" {
  region = "us-west-2"
  access_key = "AKIAVNPLXOGUAGQNP6LT"
  secret_key = "5J1CWJvqwFQh50sT5u789ndFvmWc+X2bllf96neJ"
}
resource "aws_vpc" "example_vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name = "example-vpc"
  }
}

resource "aws_instance" "example_instance" {
  ami          = "ami-08f7912c15ca96832"
  instance_type = "t2.micro"
  key_name     = "WOW"
}

```

```
subnet_id    = aws_subnet.example_subnet.id
tags = {
    Name = "example-instance"
}
}
```