Infrastructure As A Code using Terraform

By: Salma Salah
Guidance & Support : Eng. Saad El-Kenawy

Why IAAC

1.
Increased consistency and execution speed

2.
Reusing code for infrastructure replication

3.
Validating code before executing using code review and test cases

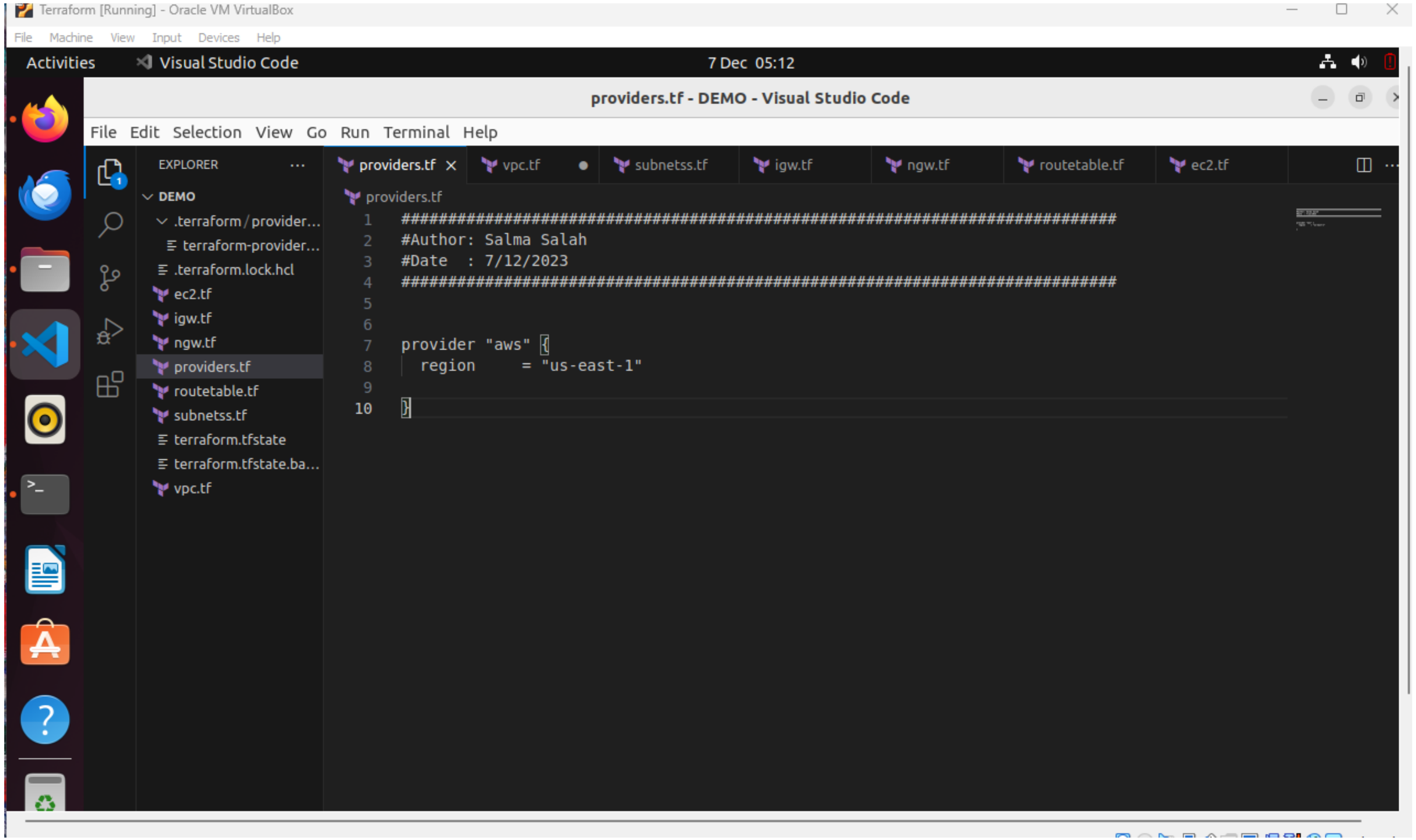
4.
Versioning the IaC helps us to rollback to the previous version

Task Scenario

Create a simple Highly Available and Secured Network Architecture with Two Public Subnets; Two Private subnets . Internet Gateway , Nat Gateway, Route Tables and an EC2 instance.

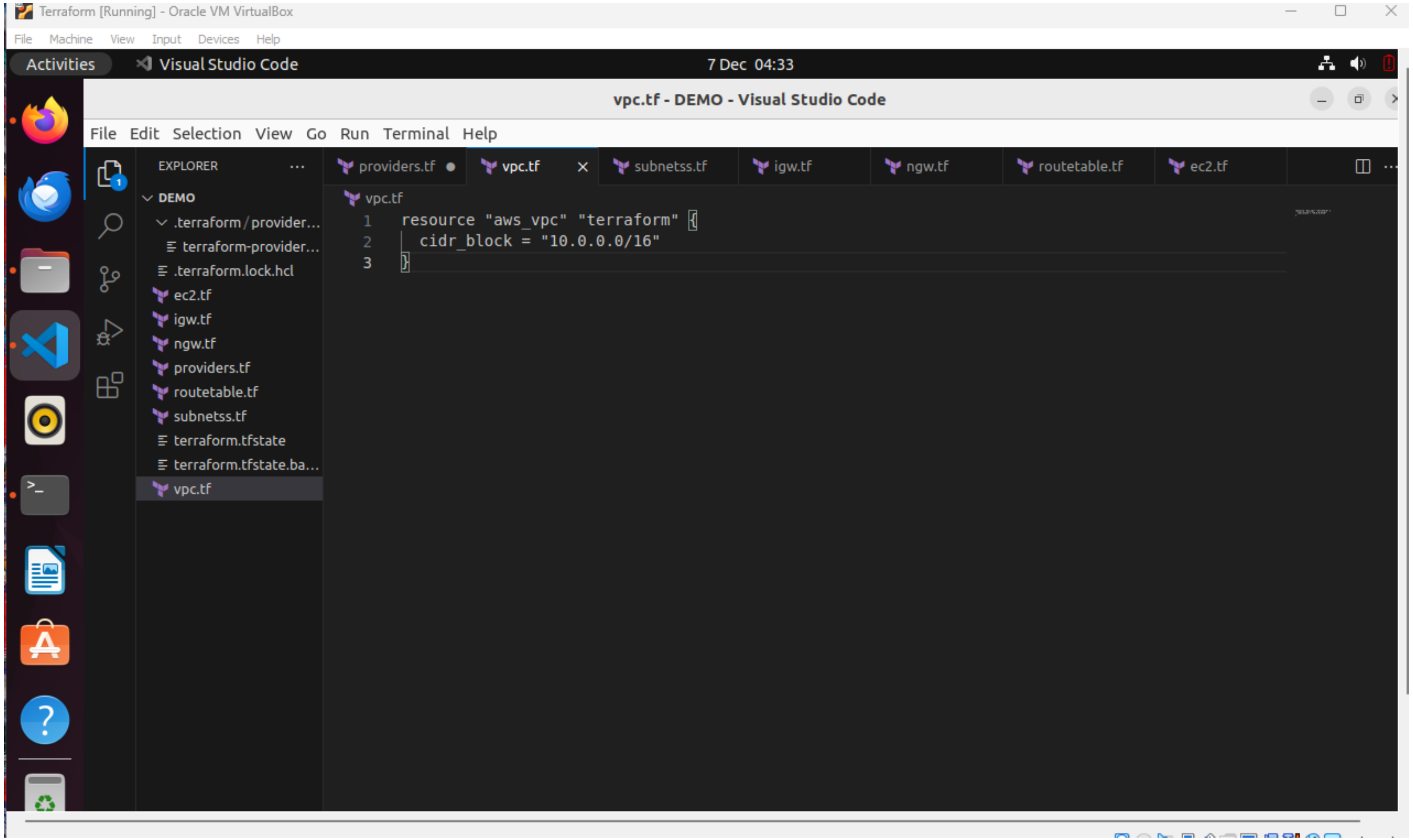
Using Terraform

1) Providers



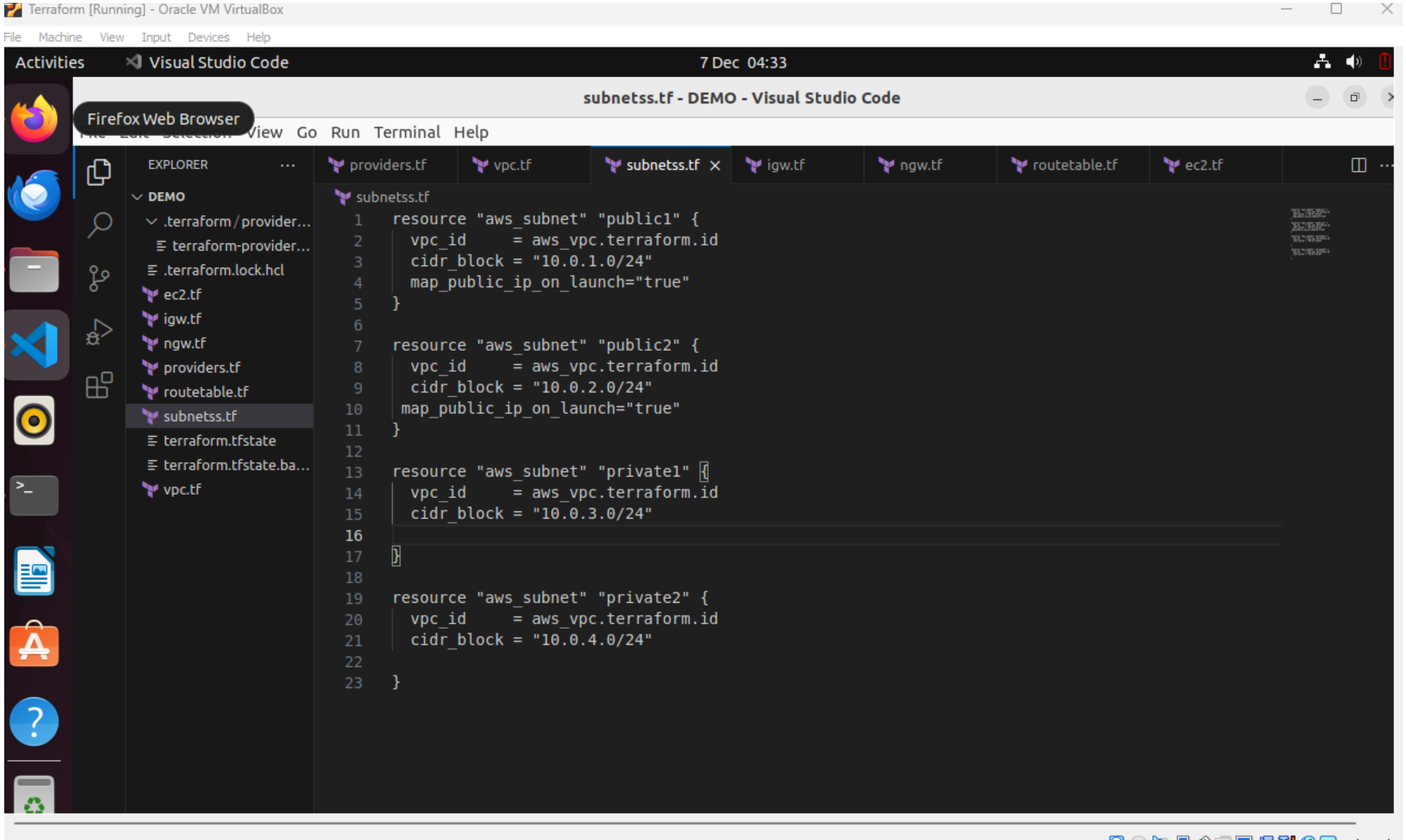
```
###############################################################################
#Author: Salma Salah
#Date  : 7/12/2023
###############################################################################


provider "aws" {
  region     = "us-east-1"

}
```

2) VPC



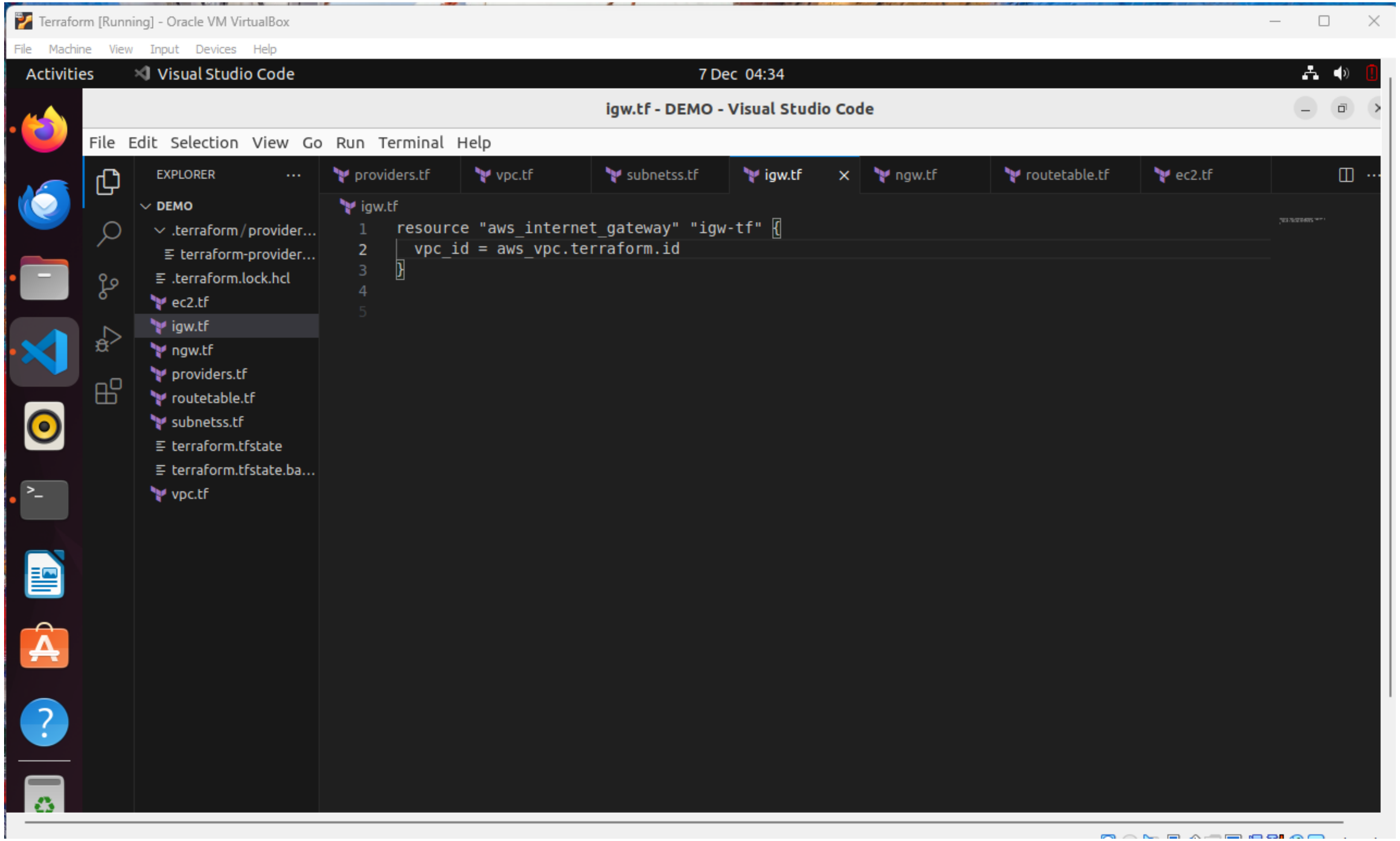
```
resource "aws_vpc" "terraform" {
  cidr_block = "10.0.0.0/16"
}
```

3) Two Public subnets & Twp Private subnets



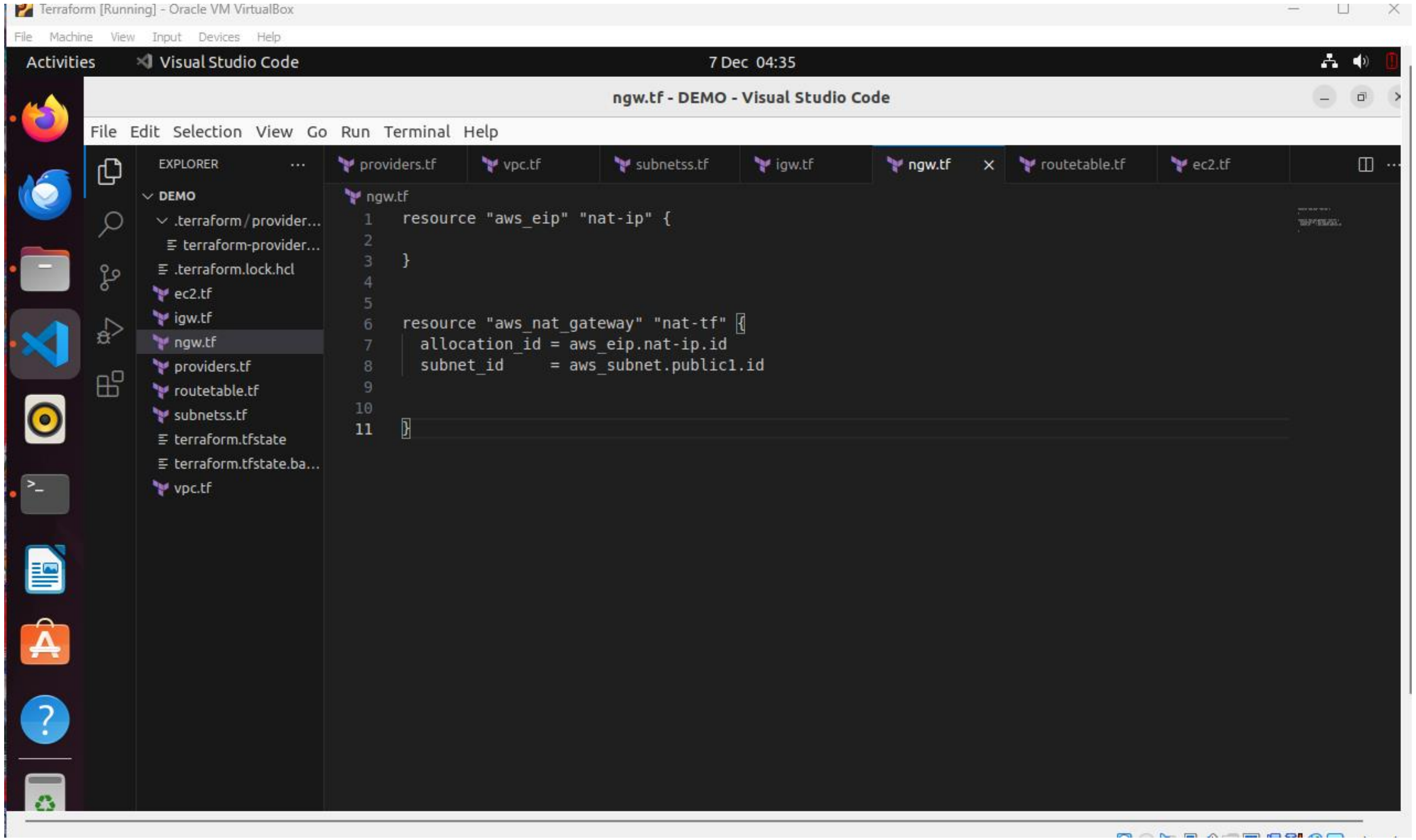
```
resource "aws_subnet" "public1" {
  vpc_id     = aws_vpc.terraform.id
  cidr_block = "10.0.1.0/24"
  map_public_ip_on_launch="true"
}

resource "aws_subnet" "public2" {
  vpc_id     = aws_vpc.terraform.id
  cidr_block = "10.0.2.0/24"
 map_public_ip_on_launch="true"
}

resource "aws_subnet" "private1" {
  vpc_id     = aws_vpc.terraform.id
  cidr_block = "10.0.3.0/24"

}

resource "aws_subnet" "private2" {
  vpc_id     = aws_vpc.terraform.id
  cidr_block = "10.0.4.0/24"

}
```

4) Internet Gateway



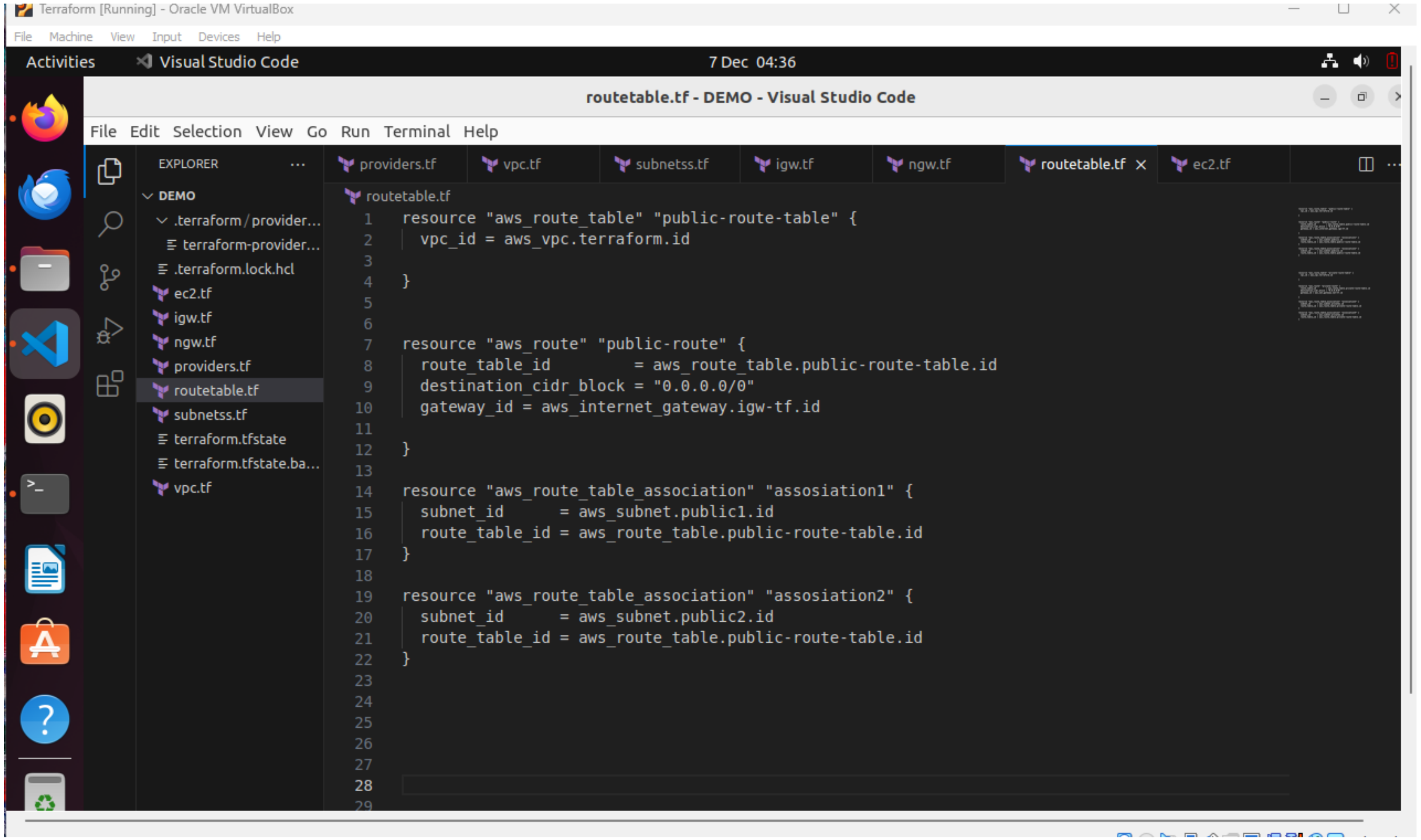
```
resource "aws_internet_gateway" "igw-tf" {
  vpc_id = aws_vpc.terraform.id
}
```

5) Nat Gateway



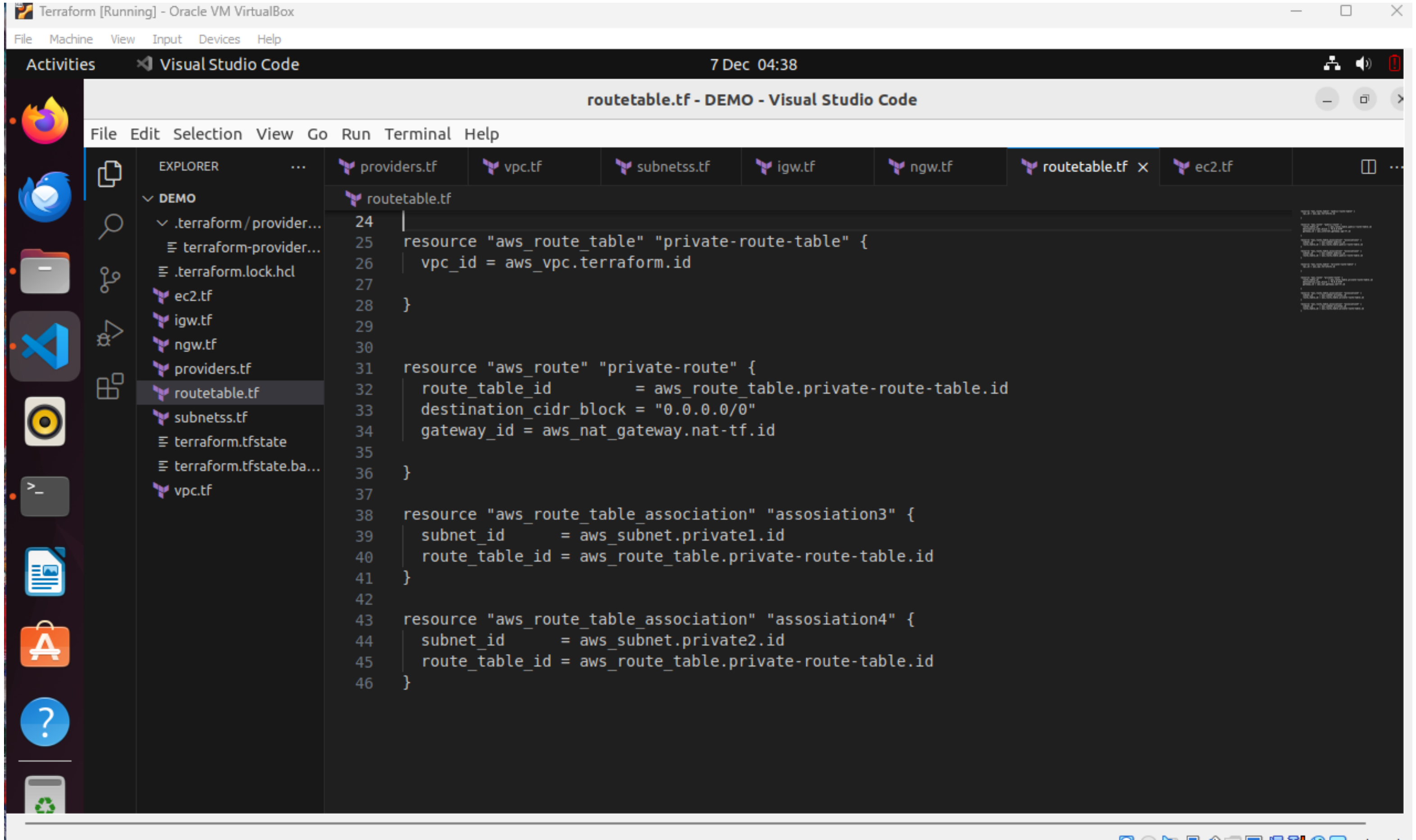
```
resource "aws_eip" "nat-ip" {

}


resource "aws_nat_gateway" "nat-tf" {
  allocation_id = aws_eip.nat-ip.id
  subnet_id     = aws_subnet.public1.id


}
```

6) Public Route Table & Subnet Assosiation


Terraform [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Visual Studio Code
7 Dec 04:36
routetable.tf - DEMO - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
DEMO
.terraform / provider...
terraform-provider...
.terraform.lock.hcl
ec2.tf
igw.tf
ngw.tf
providers.tf
routetable.tf
subnetss.tf
terraform.tfstate
terraform.tfstate.ba...
vpc.tf
providers.tf vpc.tf subnetss.tf igw.tf ngw.tf routetable.tf ec2.tf

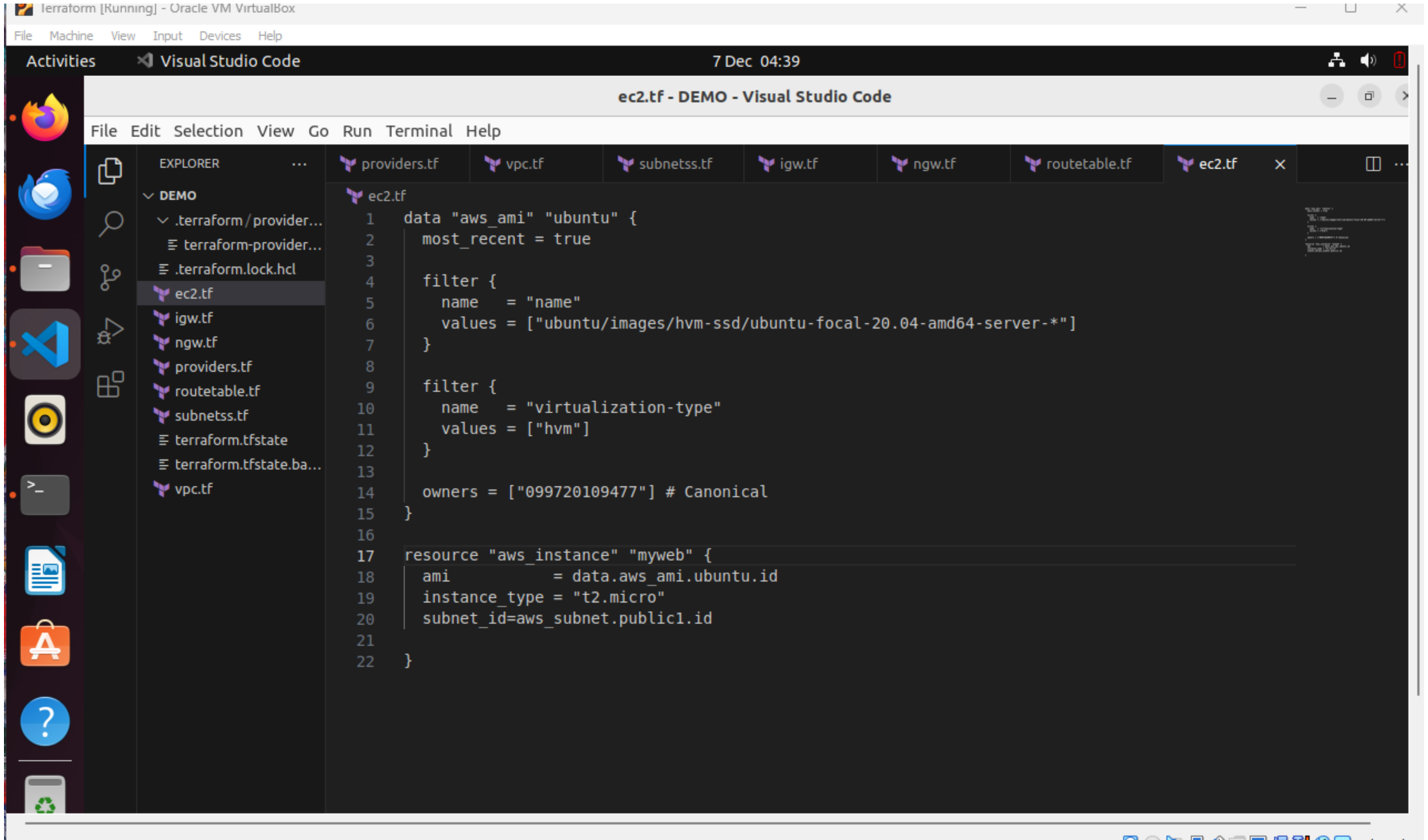
```
resource "aws_route_table" "public-route-table" {
  vpc_id = aws_vpc.terraform.id

}


resource "aws_route" "public-route" {
  route_table_id         = aws_route_table.public-route-table.id
  destination_cidr_block = "0.0.0.0/0"
  gateway_id = aws_internet_gateway.igw-tf.id

}

resource "aws_route_table_association" "assosiation1" {
  subnet_id      = aws_subnet.public1.id
  route_table_id = aws_route_table.public-route-table.id
}

resource "aws_route_table_association" "assosiation2" {
  subnet_id      = aws_subnet.public2.id
  route_table_id = aws_route_table.public-route-table.id
}
```

7) Private Route Table & Subnet Association



```
resource "aws_route_table" "private-route-table" {
  vpc_id = aws_vpc.terraform.id

}


resource "aws_route" "private-route" {
  route_table_id         = aws_route_table.private-route-table.id
  destination_cidr_block = "0.0.0.0/0"
  gateway_id = aws_nat_gateway.nat-tf.id

}

resource "aws_route_table_association" "assosiation3" {
  subnet_id      = aws_subnet.private1.id
  route_table_id = aws_route_table.private-route-table.id
}

resource "aws_route_table_association" "assosiation4" {
  subnet_id      = aws_subnet.private2.id
  route_table_id = aws_route_table.private-route-table.id
}
```

8) EC2 instance



```
data "aws_ami" "ubuntu" {
  most_recent = true

  filter {
    name   = "name"
    values = ["ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-*"]
  }

  filter {
    name   = "virtualization-type"
    values = ["hvm"]
  }

  owners = ["099720109477"] # Canonical
}

resource "aws_instance" "myweb" {
  ami           = data.aws_ami.ubuntu.id
  instance_type = "t2.micro"
  subnet_id=aws_subnet.public1.id

}
```

Used Terraform commands

1) terraform init: To Prepare working directory for other commands
2) terraform validate: To Check whether the configuration is valid
3) terraform plan: To Show changes required by the current configuration
4) terraform apply: To Create or update infrastructure
5) terraform destroy: To Destroy previously-created infrastructure