Q. Create VPC ,2 subnet 1st private and 2nd public, 2 route table and one internet gateway using terraform.

Main.tf

```
resource "aws_vpc" "my-vpc" {

 cidr_block = "10.0.0.0/16"

 enable_dns_support = true

 enable_dns_hostnames = true

}
resource "aws_security_group" "this_sg" {

 vpc_id = aws_vpc.my-vpc.id

 name   = "this_sg"

 ingress {

  from_port   = 80

  to_port     = 80

  protocol    = "TCP"

  cidr_blocks = ["0.0.0.0/0"]

 }

 ingress {

  from_port   = 443

  to_port     = 443

  protocol    = "TCP"

  cidr_blocks = ["0.0.0.0/0"]

 }

 ingress {

  from_port   = 22

  to_port     = 22

  protocol    = "TCP"

  cidr_blocks = ["0.0.0.0/0"]

 }

 ingress {
```

```hcl
    from_port   = 8080
    to_port     = 8080
    protocol    = "TCP"
    cidr_blocks = ["0.0.0.0/0"]
  }
  ingress {
    from_port   = 3306
    to_port     = 3306
    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"]
  }
}


resource "aws_subnet" "public" {
  vpc_id            = aws_vpc.my-vpc.id
  cidr_block        = "10.0.1.0/24"
  availability_zone = "us-east-1a"
  tags = {
    Name = "public"
  }
  map_public_ip_on_launch = true
}


resource "aws_subnet" "private" {
  vpc_id            = aws_vpc.my-vpc.id
```

```hcl
  cidr_block      = "10.0.2.0/24"

  availability_zone = "us-east-1b"

  tags = {

    Name = "private"

  }

  map_public_ip_on_launch = false

}
resource "aws_subnet" "private2" {

  vpc_id          = aws_vpc.my-vpc.id

  cidr_block      = "10.0.3.0/24"

  availability_zone = "us-east-1c"

  tags = {

    Name = "private2"

  }

  map_public_ip_on_launch = false

}
resource "aws_internet_gateway" "this_ig" {

  vpc_id = aws_vpc.my-vpc.id

  tags = {

    Name = "this_ig"

  }

}


resource "aws_network_interface" "ninter" {

  subnet_id = aws_subnet.public.id

  tags = {

    Name = "ninter"

  }

}
resource "aws_route_table" "this_rt" {

  vpc_id = aws_vpc.my-vpc.id
```

```hcl
  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.this_ig.id
    }


  tags = {
    Name = "this_rt"
   }
}
resource "aws_route_table_association" "route_association" {
    subnet_id     = aws_subnet.public.id
    route_table_id = aws_route_table.this_rt.id
  }
resource "aws_instance" "this_instance" {
  ami             = var.aws_ami
  instance_type       = var.aws_instance_type
  vpc_security_group_ids = [aws_security_group.this_sg.id]
  subnet_id         = aws_subnet.public.id
  tags = {
    Name = "this_instance"
  }
  root_block_device {
    volume_size = var.volume_size
  }
  user_data       = <<-EOF
    #!/bin/bash
    sudo yum update -y
    sudo wget -O /etc/yum.repos.d/jenkins.repo \
    https://pkg.jenkins.io/redhat-stable/jenkins.repo
    sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
    sudo yum upgrade
```

```hcl
    sudo dnf install java-17-amazon-corretto -y

    sudo yum install jenkins -y

    sudo systemctl enable jenkins

    sudo systemctl start jenkins

    EOF

}

resource "aws_db_subnet_group" "subnet_group" {

  name       = "subnet_group"

  subnet_ids = [aws_subnet.private.id, aws_subnet.private2.id]

  tags = {

    Name = "subnet_group"

  }

}

resource "aws_db_instance" "this_db" {

  allocated_storage = var.volume_size

  instance_class    = var.aws_db_instance_class

  engine            = var.aws_db_engine

  engine_version    = var.aws_db_engine_version

  username          = var.aws_db_master_username

  password          = var.aws_db_master_user_password

  port              = var.aws_db_port

  storage_type      = var.aws_db_storage_type

  tags = {

    Name = var.aws_db_name

  }

  vpc_security_group_ids = [aws_security_group.this_sg.id]

  db_subnet_group_name   = aws_db_subnet_group.subnet_group.name

}

resource "aws_instance" "apache" {

  ami               = var.aws_ami

  instance_type     = var.aws_instance_type
```

```
  vpc_security_group_ids = [aws_security_group.this_sg.id]

  subnet_id          = aws_subnet.public.id

  tags = {

    Name = "apache"

  }

  root_block_device {

    volume_size = var.volume_size

  }

}
```

Vars.tf

```
variable "aws_instance_type" {

  type    = string

  default = "t2.micro"


}


variable "aws_ami" {

  type    = string

  default = "ami-0440d3b780d96b29d"

}

variable "volume_size" {

  type    = number

  default = 10

}

variable "aws_db_engine" {

  type    = string

  default = "mysql"

}

variable "aws_db_engine_version" {

  type    = string
```

```hcl
  default = "5.7"
}

variable "aws_db_instance_class" {
  type    = string
  default = "db.t2.micro"

}

variable "aws_db_master_user_password" {
  type    = string
  default = "87654321"

}

variable "aws_db_master_username" {
  type    = string
  default = "myvpc"

}


variable "aws_db_name" {
  type    = string
  default = "mydb"

}

variable "aws_db_port" {
  type    = string
  default = "3306"
}

variable "aws_db_storage_type" {
  type    = string
  default="gp2"
```

}