**This is for learning references only**

**Terraform**

This configuration creates:A VPC with a CIDR block of 10.0.0.0/16.A public subnet within the VPC with CIDR block 10.0.1.0/24.A security group allowing inbound traffic on port 22 (SSH) from anywhere.Two EC2 instances in the public subnet with a specified AMI and instance type.A security group attached to the EC2 instances allowing inbound SSH traffic.A load balancer with an application load balancer type, associated with the public subnet.The instances are attached to the load balancer target group to distribute traffic.The public DNS of the instances is outputted for reference.Please make sure to replace placeholders such as AMI IDs, availability zones, and other configuration details with your specific values. Additionally, ensure that you have the necessary permissions and AWS credentials configured to apply this Terraform configuration.

# Define provider

provider "aws" {

region = "us-east-1" # Update with your desired region

}

# Create VPC

resource "aws\_vpc" "my\_vpc" {

cidr\_block = "10.0.0.0/16"

}

# Create public subnet

resource "aws\_subnet" "public\_subnet" {

vpc\_id = aws\_vpc.my\_vpc.id

cidr\_block = "10.0.1.0/24"

availability\_zone = "us-east-1a" # Update with your desired availability zone

map\_public\_ip\_on\_launch = true

}

# Create security group for EC2 instances

resource "aws\_security\_group" "instance\_sg" {

vpc\_id = aws\_vpc.my\_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"]

}

}

# Create EC2 instances

resource "aws\_instance" "ec2\_instance" {

count = 2

ami = "ami-12345678" # Update with your desired AMI

instance\_type = "t2.micro" # Update with your desired instance type

subnet\_id = aws\_subnet.public\_subnet.id

security\_groups = [aws\_security\_group.instance\_sg.id]

tags = {

Name = "EC2-instance-${count.index}"

}

}

# Create load balancer

resource "aws\_lb" "my\_lb" {

name = "my-load-balancer"

internal = false

load\_balancer\_type = "application"

security\_groups = [aws\_security\_group.instance\_sg.id]

subnets = [aws\_subnet.public\_subnet.id]

enable\_deletion\_protection = false

tags = {

Name = "MyLoadBalancer"

}

}

# Attach instances to the load balancer

resource "aws\_lb\_target\_group\_attachment" "ec2\_attachment" {

count = 2

target\_group\_arn = aws\_lb.my\_lb.arn

target\_id = aws\_instance.ec2\_instance[count.index].id

port = 80

}

# Output

output "public\_dns" {

value = aws\_instance.ec2\_instance[\*].public\_dns

}