provider "aws" {

  region = "ap-south-1" # Replace with your desired AWS region

}

resource "aws\_instance" "httpd\_instance" {

  ami           = "ami-0614680123427b75e" # Amazon Linux 2 AMI (replace if needed)

  instance\_type = "t2.micro"

  # Configure user data to install httpd

  user\_data = <<-EOF

              #!/bin/bash

              yum update -y

              yum install -y httpd

              systemctl start httpd

              systemctl enable httpd

              echo "<h1>Welcome to Apache Server on EC2!</h1>" > /var/www/html/index.html

              EOF

  tags = {

    Name = "httpd-instance"

  }

  # Assign a key pair if needed

  key\_name = "example" # Replace with your key pair name

  # Add a security group for HTTP access

  vpc\_security\_group\_ids = [aws\_security\_group.http\_access.id]

}

resource "aws\_security\_group" "http\_access" {

  name        = "http-access"

  description = "Allow HTTP access"

  ingress {

    from\_port   = 80

    to\_port     = 80

    protocol    = "tcp"

    cidr\_blocks = ["0.0.0.0/0"] # Allow access from all IPs

  }

  egress {

    from\_port   = 0

    to\_port     = 0

    protocol    = "-1" /\* Allow all outbound traffic\*/

    cidr\_blocks = ["0.0.0.0/0"]

  }

}

terraform {

  backend "s3" {

    bucket         = "s3backend"  /\*Replace with your bucket name\*/

    key            = "terraform/state.tfstate"

    region         = "ap-south-1"

    encrypt        = true

    dynamodb\_table = "terraform-state-lock"

  }

}

# terraform {

#   backend "s3" {

#     bucket         = "my-terraform-state-bucket"  # Replace with your bucket name

#     key            = "terraform/state.tfstate"

#     region         = "ap-south-1"

#     encrypt        = true                          /\*Enable encryption for state file\*/

#     dynamodb\_table = "terraform-state-lock"       /\*DynamoDB table name for state locking\*/

#   }

# }