**Project info:**

**Project statement: Implementing Load Balancing with Elastic load Balancer**

**Description: Set up a load balancer to distribute Incoming traffic across multiple Ec2 instances**

**Team Name: cloud 5**

Steps followed:

**Step 1: Creating Instances**

(2 different instances in 2 different subnet to check high availability)

GO to ec2 management console > launch instances

Name: Instance 1

AMI: Amazon Linux

Instance type: t2 micro

Key pair: create a new key pair

Network setting: select subnet 1a

Security group: create a new security group (allow Ssh and http from anywhere IPV4)

Advance details >user data (this is optional, if needed copy paste the entire script)

#!/bin/bash

yum update -y

yum install -y httpd

# Fetch the public IP address of the instance

PUBLIC\_IP=$(ec2-metadata -o | cut -d " " -f 2)

echo "Hello, World! This instance's public IP address is: ${PUBLIC\_IP}" > /var/www/html/index.html

systemctl start httpd

systemctl enable httpd

**Step 3: creating instance 2**

Create another instance with same configuration as above, but choose the different subnet, say south 1b)

**Step 4: Creating load balancer**

Go to load balancer in ec2 console >create load balancer >application load balancer

Name: Demo load balancer

Keep other settings as default

Select all the mappings

Create a new security group (allow Ssh and http from anywhere IPV4) and attach it

Create a new target group>choose instance type

Name: give any name of your choice

Register targets>select both instance>include as pending now

Create a target group and attach this to your load balancer >create a load balancer

**Step 5: Testing and verification**

For testing this project select the load balancer which you have created

Copy the DNS name of load balancer

Go to any browser >paste the DNS name >refresh the page

When you refresh the page you must see two different instance address after every refresh, it shows that load balancer is switching server loads to different instance

Load balancers keep any one server from becoming overloaded. They ensure that no one server is overworked, which could degrade performance

**Step 6: Health check**

To check health of instances purposely terminate any one instance,

Go to target group >check the register instance

You can see that one instance is unused.

So these are the steps I carried out to complete this project