

# **GLA UNIVERSITY (MATHURA)**



## **Computer Science And Engineering Department**

**COURSE - JOVAC AWS**

**SUPERVISOR - Mr. Raushan Kumar Singh  
Dr. Ruby Panwar**



**TOPIC :-**

# **Create an AutoScaling Group and put Load Balancer, Test it.**

**Presented By:-**

**KULDEEP SHARMA (2215000956)**

**ANAMIKA SINGH (2215000221)**

**TANISH GUPTA (2215001829)**

**VISHNU UPADHYAY (2215001991)**

**SANDEEP KUMAR (2215001573)**



# STEPS FOR CREATING PROJECT ON AWS.

- Create **two server/instances**.
- Create **target group & connect server**.
- Create **Load Balancer**.
- Create **Launch template**.
- Create **Autoscaling group**.



# What is Autoscaling group?

- **Autoscaling Group** helps to create automatically adjusts the number of instances to handle the load if load will decrease then instances will terminate.
- Ex:- Web Application, Netflix, Amazon, etc



# What is Load Balancer ?

- A **Load Balancer** is a service that automatically distributes incoming network traffic across multiple servers or instances. The goal of a load balancer is to ensure that no single server is overwhelmed by too much traffic, thus improving the availability, reliability, and scalability of your application.

# Launch Template:

Define parameters like AMI (Amazon Machine Image), instance type, key pairs, security groups, and more, to reuse across multiple instance launches.



# Target group:

In AWS, a **Target Group** is a configuration resource associated with **Elastic Load Balancers (ELB)** that defines how traffic should be directed to a specific set of registered instances, containers, or IP address.



# Real life Example:

- E-Commerce Website (e.g. Flipkart)
- Video Streaming Platform (e.g. Netflix)
- Financial Applications (e.g. Paytm)
- Online Gaming Services (e.g. Call of Duty)



# Advantage:

- Scalability
- High Availability
- Improved Performance and User Experience
- Simplified Management and Monitoring



# Conclusion

- The combination of an **Auto Scaling Group** and a **Load Balancer** provides a robust, scalable, cost-effective, and highly available architecture for your applications. It ensures that your system can handle varying loads while maintaining consistent performance, security, and availability.





**THANK YOU!**