# NGINX Hands-On Assignment

## Title:

Set Up an HA Reverse Proxy with Load Balancing and Web Hosting

## Submission Format:

GitHub Repository with Documentation

## Objective:

Simulate a production-like high availability environment using NGINX (Open Source) and Keepalived, where:
- One set of NGINX instances act as load balancers/reverse proxies.
- Another set of NGINX instances act as backend web servers.
- VIP failover is handled by Keepalived.

## Assignment Tasks:

1. Backend Web Servers (App Tier)

- Create two Linux-based nodes (VMs or containers).

- Install and configure NGINX to serve a static HTML page.

- Each node must serve a different version of the page (e.g., "Node A" and "Node B") to verify load balancing visually.

2. Frontend Load Balancers (LB Tier)

- Create two additional nodes to act as reverse proxies.

- Install and configure NGINX to:

  - Load balance requests between the two backend nodes using round-robin.
  - Act as a reverse proxy, listening on port 80.
  - Set appropriate proxy headers and enable basic health checks.

3. Keepalived High Availability

- Configure Keepalived on the two load balancer nodes.

- Implement VRRP so a floating Virtual IP (VIP) is assigned to the active load balancer.

- On failure of the active node (e.g., by stopping NGINX), the VIP should shift to the standby node.

## README.md Must Include:

- Overview of your setup and assumptions

- Instructions to:

  - Start backend servers
  - Start load balancers
  - Test VIP and load balancing
  - Simulate failover by stopping one LB node

- Screenshots or curl output demonstrating:

  - Round-robin load balancing
  - VIP before and after failover
  - Web server responses ("Node A" / "Node B")