**Managing Services in a Swarm Cluster with adding a worker nodes to the Swarm cluster.**

This project focuses on understanding Docker Services, the core of Swarm mode. You'll learn to deploy, scale, inspect, and update a web service in a Swarm cluster. This hands-on exercise will help you grasp how Docker orchestrates containers across nodes and ensures high availability and load balancing.

**Objectives:**

* Initialize a Swarm cluster
* Create and manage a Docker service
* Scale the service to multiple replicas
* Perform rolling updates
* Inspect service configuration and container distribution
* You will create a simple Nginx-based web service deployed in Swarm mode.

1. **Initialize Docker Swarm (on the manager node):**

docker swarm init

1. **Add worker nodes to the Swarm cluster:** Run the following command on each worker node using the join-token from the manager:

docker swarm join --token <worker-token> <manager-ip>:2377

To retrieve the token from the manager node:

docker swarm join-token worker

1. **Create a Docker service:**

docker service create \

--name web-service \

--replicas 3 \

--publish published=8080,target=80 \

nginx:alpine

This command deploys a service named web-service with 3 replicas and exposes it on port 8080.

1. **Verify service creation:**

docker service ls

1. **Check running tasks (containers):**

docker service ps web-service

1. **Scale the service up to 5 replicas:**

docker service scale web-service=5

1. **Inspect the service details:**

docker service inspect web-service --pretty

1. **Perform a rolling update to a newer image:**

docker service update --image nginx:1.25-alpine web-service

1. **Check logs and container distribution:**

docker service logs web-service

1. **Access the service in a browser:** Visit http://localhost:8080 to see the default Nginx page.