**Approach 1: Manual Container Management**

**Setup & Scaling Steps:**

1. **Create a Docker user-defined bridge network** (private network):

docker network create micro-net

1. **Run containers individually attached to the network:**

docker run -d --name service-registry --network micro-net image-service-registry

docker run -d --name config-server --network micro-net image-config-server

docker run -d --name orders-1 --network micro-net image-orders

docker run -d --name inventory-1 --network micro-net image-inventory

1. **Scale by launching additional containers from the same image:**

docker run -d --name orders-2 --network micro-net image-orders

docker run -d --name inventory-2 --network micro-net image-inventory

1. **Manage ports:**
   * Internal container ports can remain the same (e.g., 8080).
   * If exposing to host, map to unique host ports per container to avoid conflicts, e.g., -p 8081:8080, -p 8082:8080.
2. **Handle service discovery and load balancing:**
   * Use service registry (e.g., Eureka) for container instances to register themselves.
   * External clients communicate via API Gateway or load balancer.
   * You must manage starting/stopping containers and scaling manually.