**E-commerce Microservices Setup Guide**

**1. Prerequisites**

**System Requirements**

* Ubuntu (or any Linux distribution)
* Docker & Docker Compose
* Java 21
* Gradle 8.5+
* PostgreSQL (optional if using Docker)
* Kafka

**Install Required Software**

**Update System Packages**

Run:

sh

CopyEdit

sudo apt update && sudo apt upgrade -y

**Install Java 21**

Run:

sh

CopyEdit

sudo apt install openjdk-21-jdk -y

Verify installation:

sh

CopyEdit

java -version

**Install Gradle 8.5**

Run:

sh

CopyEdit

wget https://services.gradle.org/distributions/gradle-8.5-bin.zip -P /tmp

sudo unzip -d /opt/gradle /tmp/gradle-8.5-bin.zip

Set up Gradle environment:

sh

CopyEdit

echo 'export PATH=/opt/gradle/gradle-8.5/bin:$PATH' >> ~/.bashrc

source ~/.bashrc

Verify installation:

sh

CopyEdit

gradle -v

**Install Docker Compose**

Install Docker Compose:

sh

CopyEdit

sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose

Verify installation:

sh

CopyEdit

docker-compose --version

**2. Setting Up the Project**

**Clone the Repository**

Run:

sh

CopyEdit

git clone https://github.com/KPkm25/ecommerce-microservices

cd ecommerce-microservices

**Build the Microservices**

Run:

sh

CopyEdit

cd \*service-name\*

gradle clean build

Without test:  
gradle clean build -x test

This will generate the necessary .jar files in build/libs/. Repeat this for each service.

**3. Running the Application**

**Start the Services with Docker Compose**

Open docker-compose and change the image tag to your dockerhub credentials.

Run:

sh

CopyEdit

docker-compose up -d --build

This will start:

* PostgreSQL (Database)
* Kafka & Zookeeper (Event Streaming)
* Microservices (User, Product, Order, Payment)

**Verify Running Containers**

Run:

sh

CopyEdit

docker ps

You should see running services.

**Check Service Logs**

To check logs for a specific service, run:

sh

CopyEdit

docker logs order-service

**5. Stopping & Restarting**

To stop all services: Run:

sh

CopyEdit

docker-compose down

To restart: Run:

sh

CopyEdit

docker-compose up -d

To rebuild everything from scratch: Run:

sh

CopyEdit

docker-compose up -d --build

**6. Troubleshooting**

**Verify PostgreSQL in Docker**

Run:

sh

CopyEdit

docker exec -it ecommerce-microservices-postgres-1 psql -U postgres

Check databases:  
\l

Check if the **ecommerce** database exists. If not, create it manually:

sql

CopyEdit

CREATE DATABASE ecommerce;

Then, list all tables:

sql

CopyEdit

\dt

**7. Kubernetes Setup**

Push the built images to dockerhub using the following commands:  
docker-compose build

docker-compose push

Now, switch to the k8s directory which has all the deployment.yaml files required to run the Kubernetes clusters.

Modify all the deployment files by specifying the image repository.

Run all the clusters with the following command:  
kubectl apply -f .

You can view your deployments by using:  
kubectl get pods

Kubectl get all -o wide