**Project Title: Online Order Management Microservices**

**Problem Statement:**

An e-commerce company wants to manage **orders** and **products** using microservices. The system should allow clients to view products, create orders, and check order status. Each service should be **independent** and registered with **Eureka Server** for service discovery.

You will create **two microservices**:

1. **Product Service** – Manage products and stock.
2. **Order Service** – Manage customer orders.

**Microservices & Entities**

**1. Product Service**

* **Entity:** Product
  + Product ID (Primary Key)
  + Name
  + Description
  + Price
  + Stock Quantity
* **APIs**
  + Create product
  + Get all products
  + Get product by ID
  + Update product
  + Delete product

**2. Order Service**

* **Entity:** Order
  + Order ID (Primary Key)
  + Product ID (reference Product Service)
  + Quantity
  + Order Status (Pending / Completed / Cancelled)
  + Customer Name
  + Order Date
* **APIs**
  + Place a new order (check stock from Product Service)
  + Get all orders
  + Get order by ID
  + Update order status
  + Delete order

**Architecture**

1. **Eureka Server**
   * Service registry for Product Service and Order Service.
   * Services register themselves and discover each other dynamically.
2. **Service Communication**
   * Order Service calls Product Service (via REST) to check stock availability before creating an order.
3. **Database**
   * Each service has its own MySQL database:
     + Product Service → products table
     + Order Service → orders table
4. **Validation**
   * Server-side validation (quantity > 0, stock availability, required fields).