You have to create 3 microservicecs

a) Customer Service
b) Product Service
c) Order Service

1) Customer Service:

- You have to implement below operations
    - Add new Customer
    - Update Customer details
    - Fetch Customer details by customer Id
    - Delete Customer by customer Id (Note: Don’t delete customer details from database. You can use flag variable (separate column in db) that indicates customer deleted his account)
    - Fetch all orders placed by Customer

- Create a Customer model class with below properties
    - customerId: Long
    - name: String
    - email: String
    - contact: String
    - addresses: List<Address>
    - orders: List<Order>

  Note: Fetch order details from order service by passing customer id.

- Create CustomerRepository using Spring Data JPA

- You can use RestTemplate / Feign Client / WebClient to call order service.

2) Product Service:

- You have to implement below operations
    - Add new product
    - Update product details
    - Fetch product details by product id
    - Fetch all products based on category
    - Fetch all products
    - Delete product by product id

- Create Product model class with below properties
    - productId: Long
    - name: String
    - category: String
    - price: Integer
    - details: Map<String, Object>

- Create ProductRepository using Spring Data JPA

3) Order Service:

- You have to implement below operations
    - Add new order
    - Fetch order details by customer id
    - Update order details (status of order) (placed, returned, cancelled, delivered)
    - Fetch order details by order id

- Create Order model class with below properties
    - orderId: Long
    - status: OrderStatus (Create OrderStatus enum with different statuses like placed, returned, cancelled, delivered)
    - product: Product
    - quantity: Integer
    - amount: Integer (Total order amount = quantity * product price)
    - customerId: Long

  Note: Fetch product details from Product Service by passing product id

- Create OrderRepository using Spring Data JPA

- You can use RestTemplate / Feign Client / WebClient to call product service.

++

Setup Eureka Server and register customer, order and product service with it.

++

Store configuration of all microservices in GitHub repo and setup config server to provide these configurations to all microservices.

++

Setup API Gateway using Spring Cloud Gateway (All requests must go through API Gateway.

++

Implement Authentication and authorization using Spring Security and OAuth 2.0

Use keycloak server for authentication and authorization.

1) Make sure that customer must be authenticated and authorized first to access his/her account and order details.

++