Spring Boot 3 Microservices - Exercises Compilation

**1. Basic Microservices Setup**

Projects: User & Order Service, Account & Loan Service, Product & Inventory Service.

Technologies: Spring Boot REST APIs, WebClient/OpenFeign, MySQL/PostgreSQL.

Each service exposes RESTful endpoints and runs independently on a unique port (configured using application.properties).

**2. Eureka Service Discovery**

Create Eureka Discovery Server with Spring Cloud Eureka.

Annotate each microservice with @EnableDiscoveryClient and configure spring.application.name.

Launch Discovery Server on port 8761 and access via http://localhost:8761.

**3. API Gateway with Routing and Logging**

Create an API Gateway using Spring Cloud Gateway.

Configure service routes with path rewriting and rate limiting.

Example route: Path=/greet-service/\*\* -> lb://greet-service.

Add a global filter to log incoming requests.

**4. Load Balancing and Circuit Breaker**

Use Spring Cloud LoadBalancer to distribute traffic among service instances.

Implement Circuit Breaker using Resilience4j to handle slow or failed services.

Use fallback methods to return alternate responses when primary services fail.

**5. Centralized Authentication and JWT Security**

Implement OAuth 2.1/OIDC login using Google with Spring Security.

Configure Resource and Authorization Servers using Spring Security and OAuth2 Resource Server.

Use JWT tokens for secure communication, create custom token filters to validate and authorize requests.