Spring 2025 – SWER007

Lab 10: Service Discovery with Eureka

✅ Objective

Set up **Eureka Service Discovery** to allow employee-service and department-service to register themselves dynamically. This removes hardcoded URLs and enables dynamic communication between services.

🔧 Prerequisites

You should now have:

- employee-service running on port 8081
- department-service running on port 8082

1 Create the Discovery Server

Use Spring Initializr or your IDE to create a new project:

🔹 discovery-service

```
spring init --dependencies=eureka-server discovery-service
```

Add @EnableEurekaServer in the main application class:

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.netflix.eureka.server.EnableEurekaServer;

@SpringBootApplication
@EnableEurekaServer
public class DiscoveryServiceApplication {
    public static void main(String[] args) {
        SpringApplication.run(DiscoveryServiceApplication.class, args);
    }
}
```

Configure application.yml:

```
server:
  port: 8761

spring:
  application:
    name: discovery-service

eureka:
  client:
    register-with-eureka: false
    fetch-registry: false
```

Run the server and open: http://localhost:8761

2 Register Services with Eureka

Add the Eureka Client dependency to both employee-service and department-service:

```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
</dependency>
```

In both application.yml files:

```
spring:
 application:
  name: employee-service  # or department-service

 # example for employee-service
 eureka:
  client:
   service-url:
    defaultZone: http://localhost:8761/eureka
```

Add @EnableDiscoveryClient:

```
@EnableDiscoveryClient
@SpringBootApplication
public class EmployeeServiceApplication { }
```

3 Test Registration

Run:

- discovery-service
- employee-service
- department-service

Go to http://localhost:8761

- You should see both services listed under **Registered Applications**

4 Clean Up Hardcoded URLs

In preparation for inter-service calls, replace hardcoded URLs with service names:

- e.g. http://localhost:8082/departments ➔ http://department-service/departments
- This will work after we add a **load balancer or gateway** in the next lab

📝 Lab Tasks

1. Create and run a new project: discovery-service
2. Configure it as a Eureka server
3. Register employee-service and department-service
4. Test registration on the Eureka dashboard
5. Replace hardcoded service URLs with logical names

🎯 Expected Outcome

- Eureka server running at localhost:8761
- employee-service and department-service are auto-registered
- Services are now discoverable dynamically