# lab3 Spring Cloud服务注册与发现

1. 搭建Eureka Server

1.创建一个新模块，eureka-server微服务

2.导入依赖

1）在父工程导入Spring Cloud的依赖

<!-- 定义SpringCloud版本 -->  
<repositories>  
 <repository>  
 <id>spring-snapshots</id>  
 <name>Spring Snapshots</name>  
 <url>https://repo.spring.io/snapshot</url>  
 <snapshots>  
 <enabled>true</enabled>  
 </snapshots>  
 </repository>  
 <repository>  
 <id>spring-milestones</id>  
 <name>Spring Milestones</name>  
 <url>https://repo.spring.io/milestone</url>  
 <snapshots>  
 <enabled>false</enabled>  
 </snapshots>  
 </repository>  
</repositories>

<!-- 锁定SpringCloud版本 -->  
<dependencyManagement>  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-dependencies</artifactId>  
 <version>Finchley.M9</version>  
 <type>pom</type>  
 <scope>import</scope>  
 </dependency>  
 </dependencies>  
</dependencyManagement>

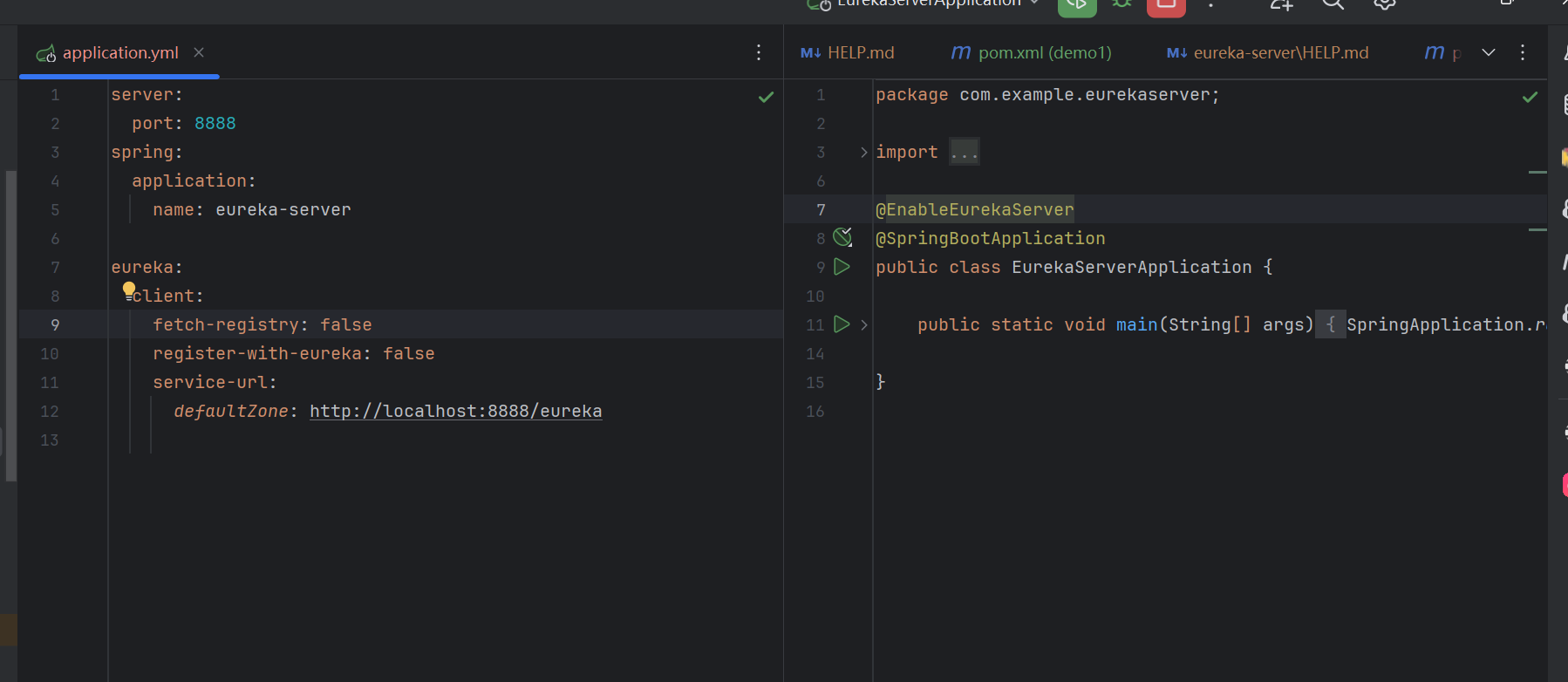
1. 在eureka-server子工程导入Eureka Server的依赖

<!-- Eureka服务器-->  
<dependencies>  
 <dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>  
 </dependency>  
</dependencies>

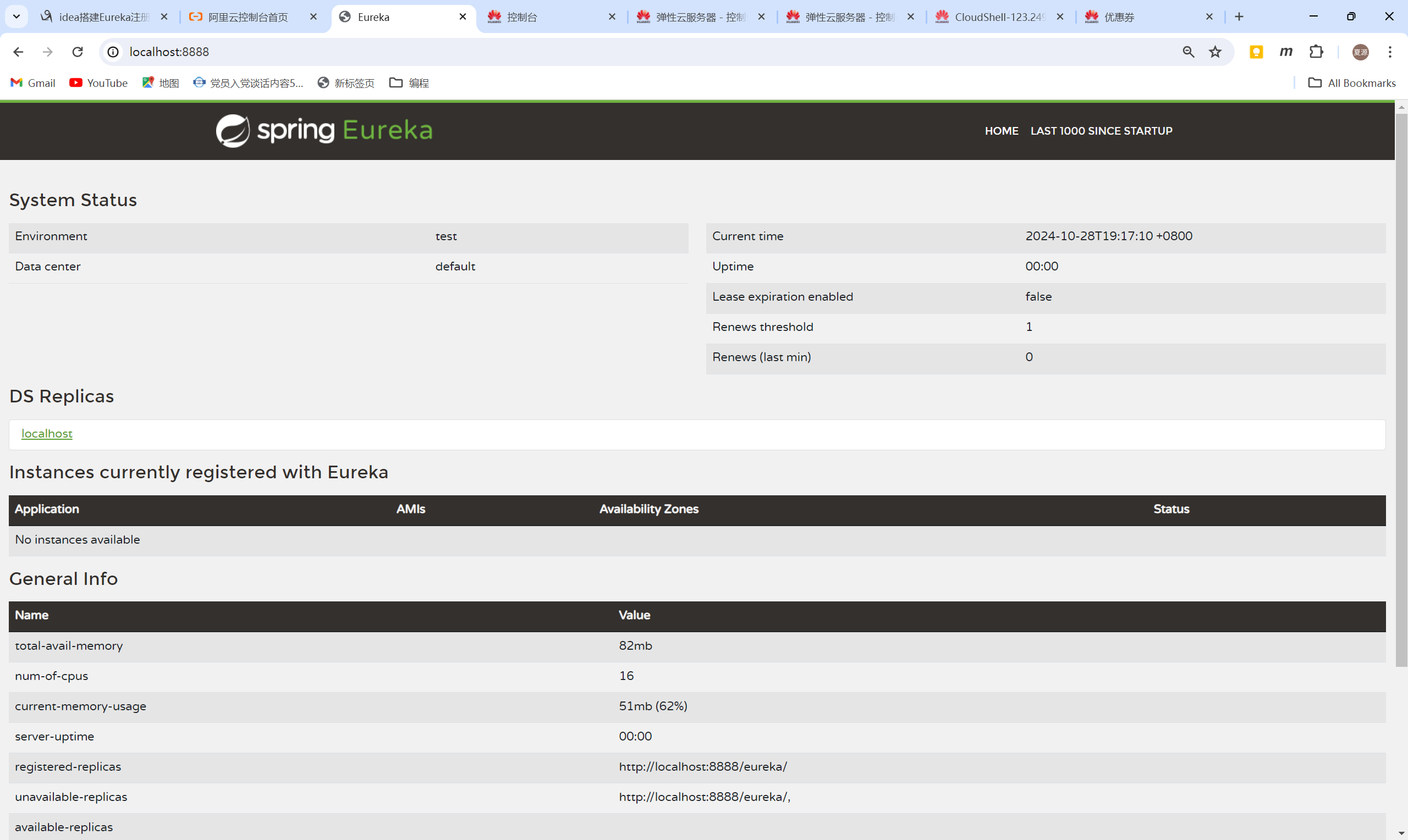
1. 编写application.yml配置Eureka

server:  
 port: 8888  
spring:  
 application:  
 name: eureka-server  
 # 单机版配置  
eureka:  
 client:  
 fetch-registry: false # 是否需要从Eureka获取注册信息  
 register-with-eureka: false # 是否需要把该服务注册到Eureka  
 service-url: # 暴露Eureka注册地址  
 defaultZone: http://localhost:8888/eureka

1. 编写启动类（添加@EnableEurekaServer注解，其余代码同前）



1. 启动Eureka，访问Eureka（浏览器输入locallhost:8888)，检查Eureka是否启动成功



1. 搭建Eureka Client（所有微服务均按以下步骤注册到Eureka）
2. 导入Eureka Client的依赖

<!-- 导入Eureka客户端依赖-->  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>  
</dependency>

1. 在application.yml配置连接Eureka Server

eureka:

client:

register-with-eureka: true # 作为客户端需要注册到Eureka

fetch-registry: true # 作为客户端需要从Eureka获取注册信息

service-url: # 客户端注册地址

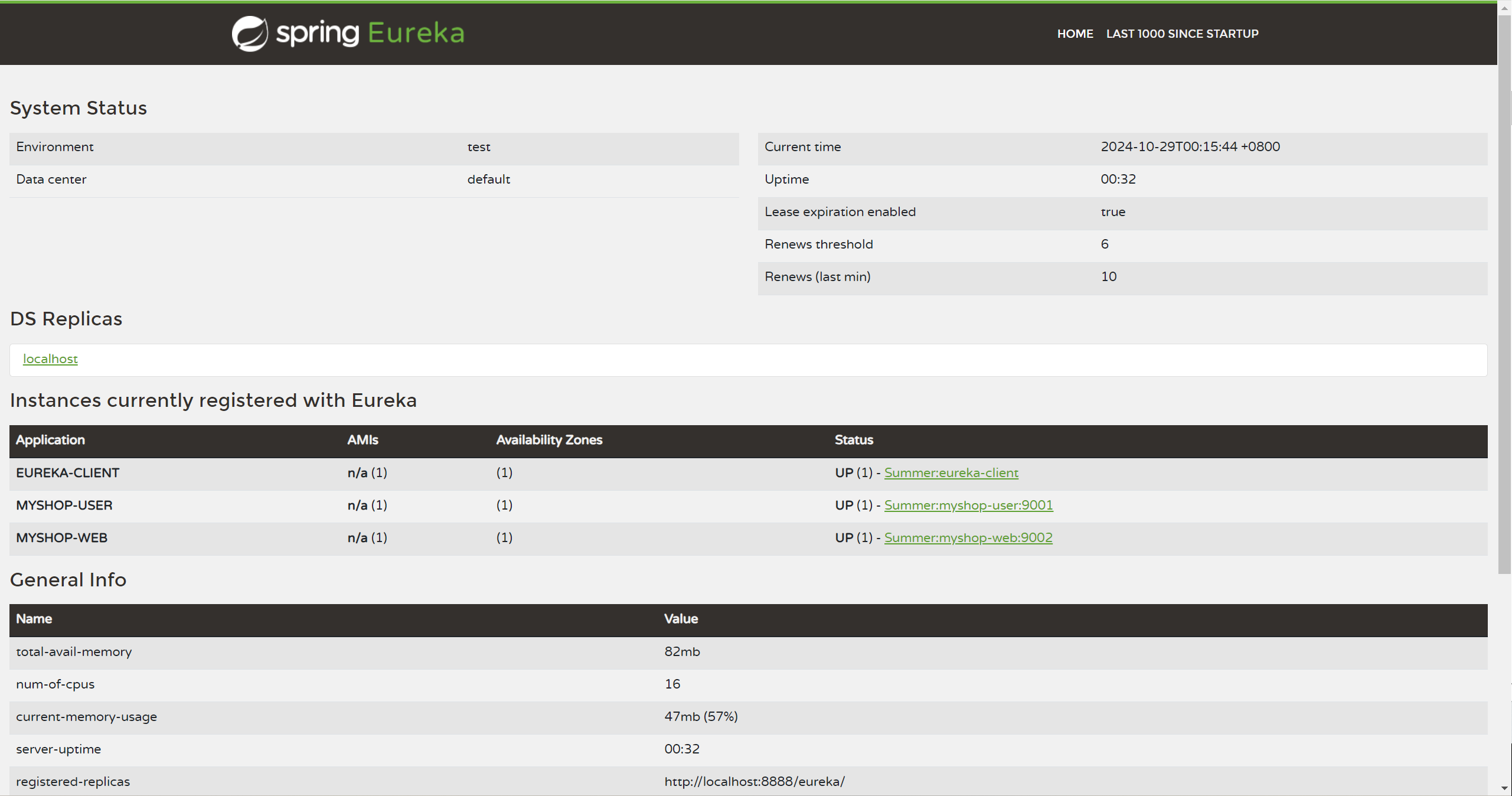
defaultZone: http://localhost:8888/eureka

instance:

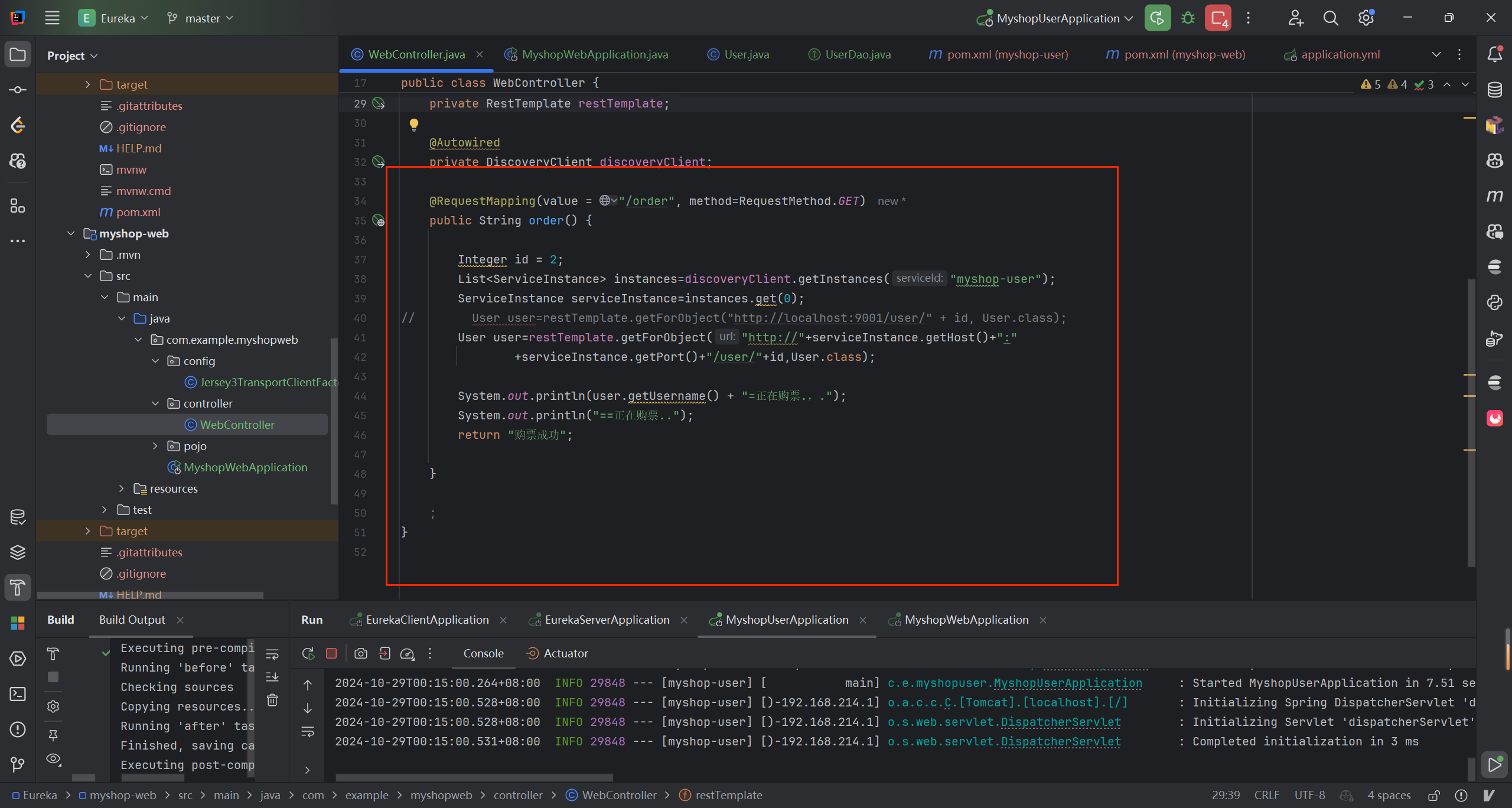
# 优先使用该服务的IP地址注册到Eureka，在生产环境建议改为true

prefer-ip-address: true

1. 在启动类添加@EnableEurekaClient注解
2. 重启微服务，观察记录Eureka界面内容（用户微服务及购票微服务均需注册至Eureka）

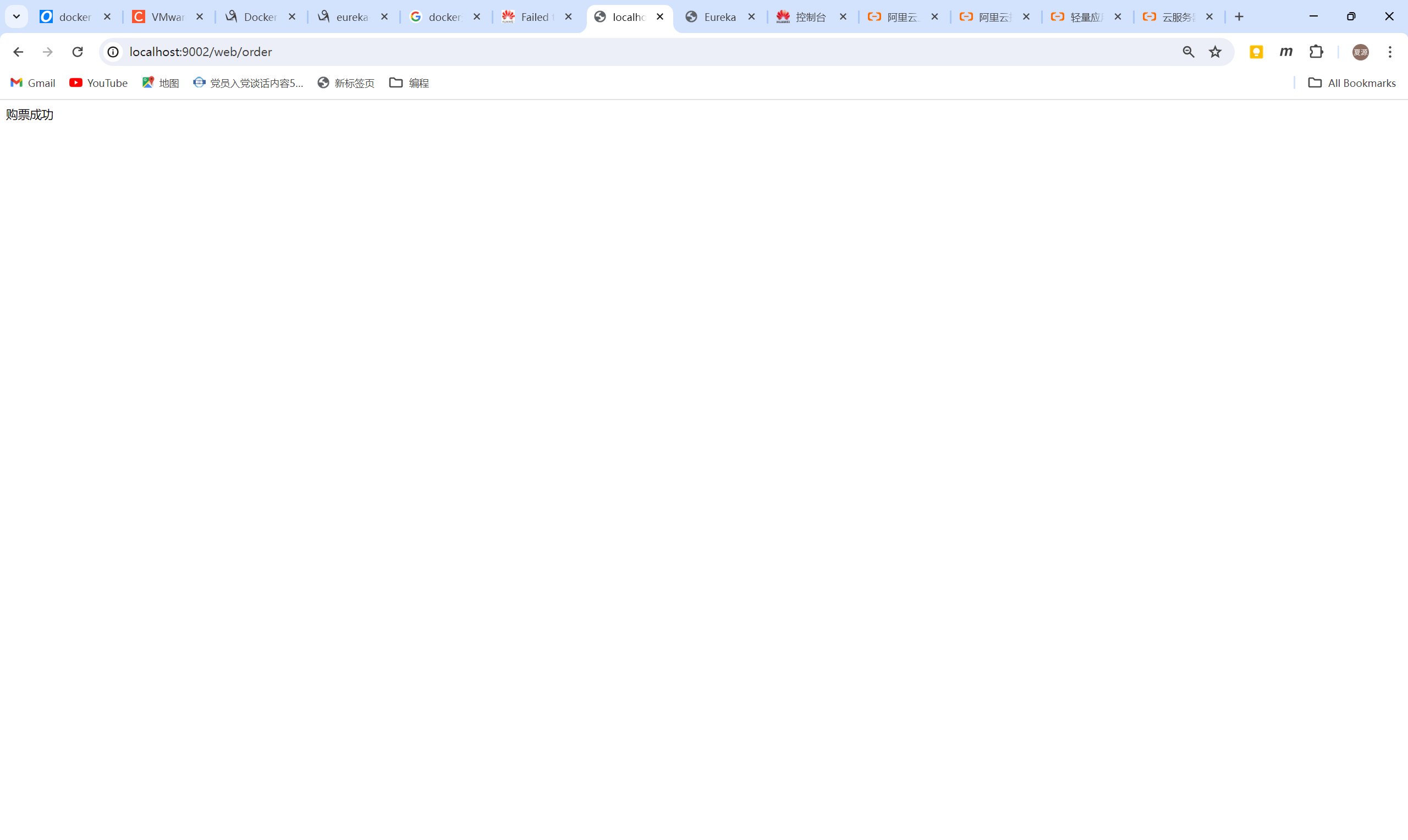


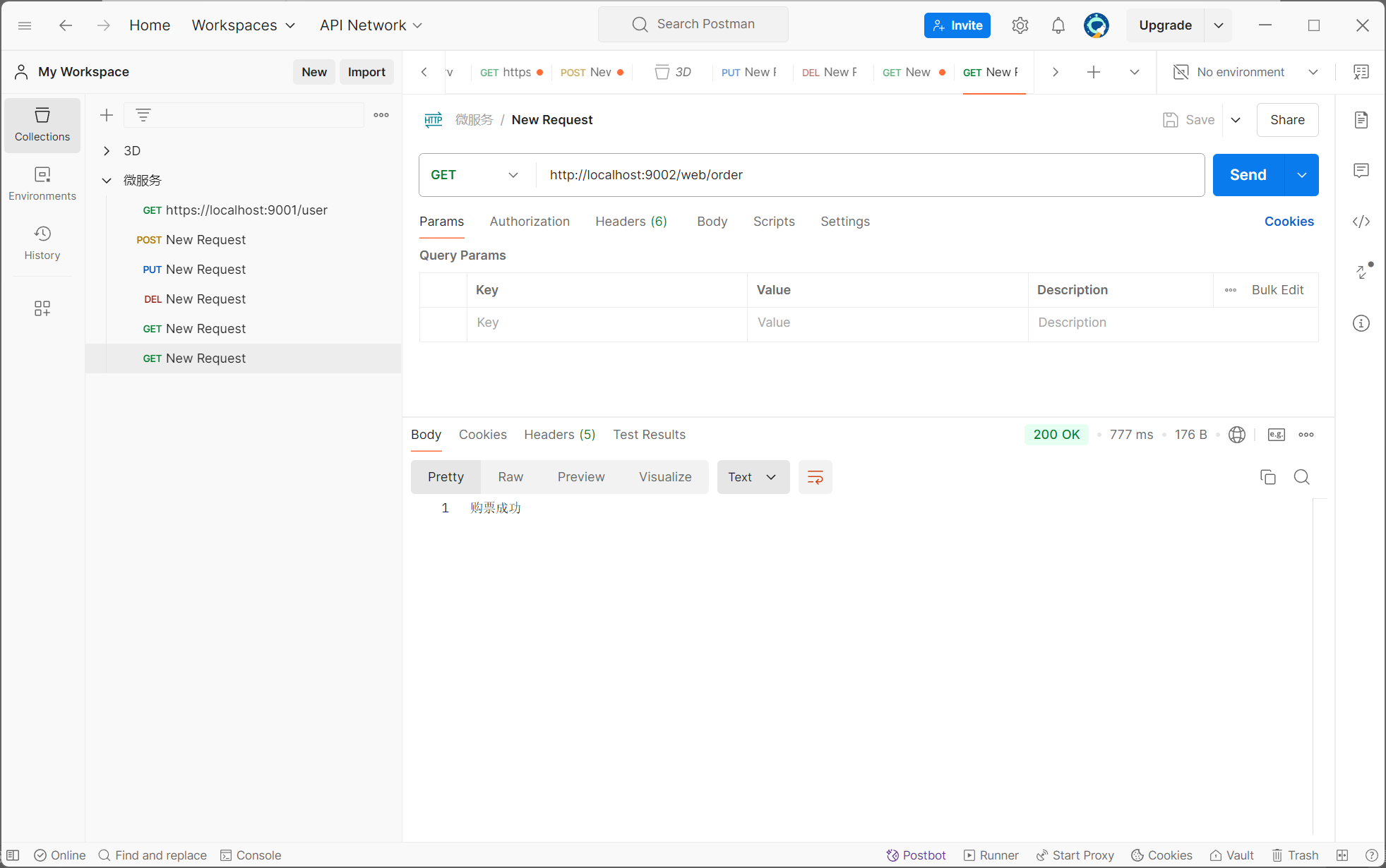
1. 使用Eureka实现服务调用
2. 修改购票微服务的order（）方法：注入自动发现客户端对象discoveryClient，通过discoveryClient.getInstances方法获得与微服务名匹配的服务列表，并调用服务列表中第一个服务的相应方法。

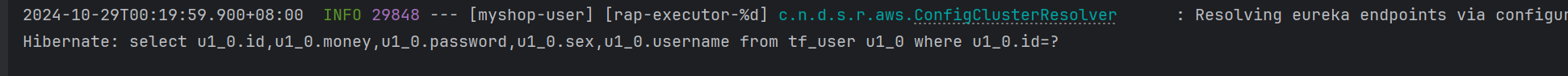


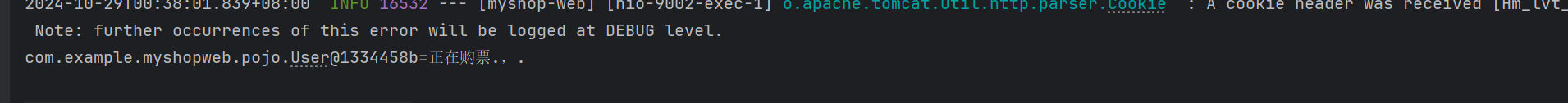


1. 重启购票微服务，浏览器上请求调用order方法（localhost:9002/web/order)，查看并记录请求返回结果及IDEA控制台中用户微服务及购票微服务信息。







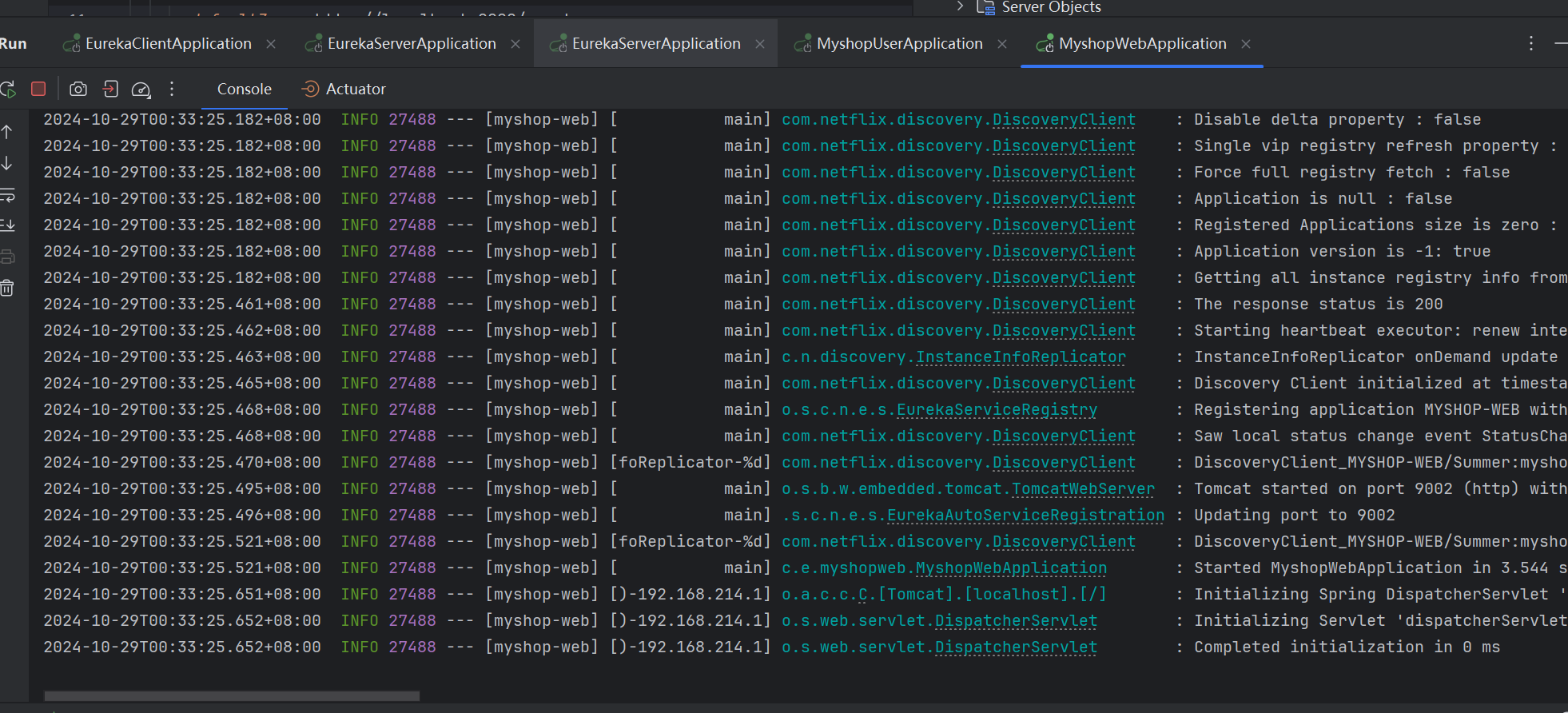


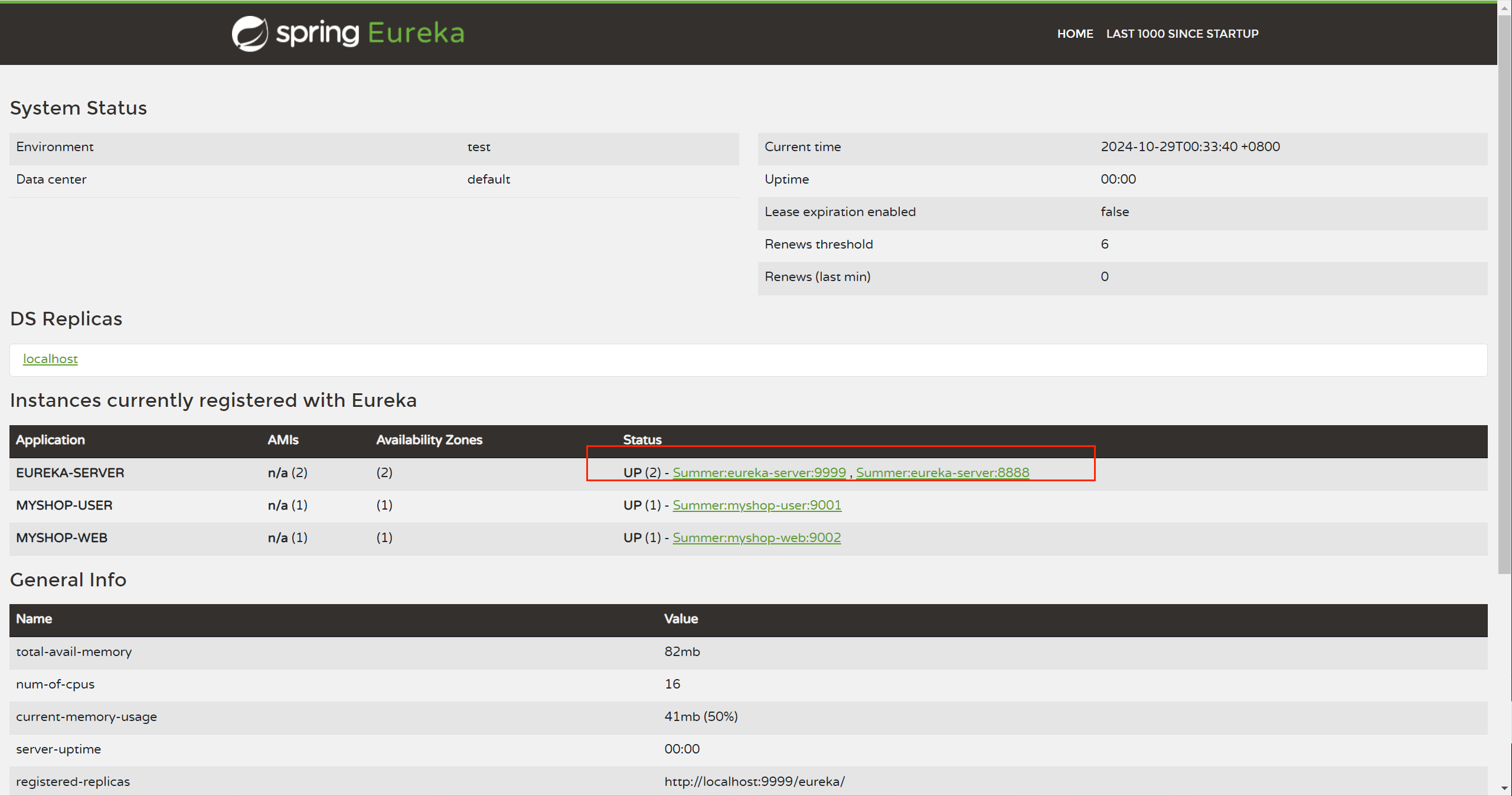


1. 搭建高可用的Eureka Server
2. 修改application.yml配置文件，如下：



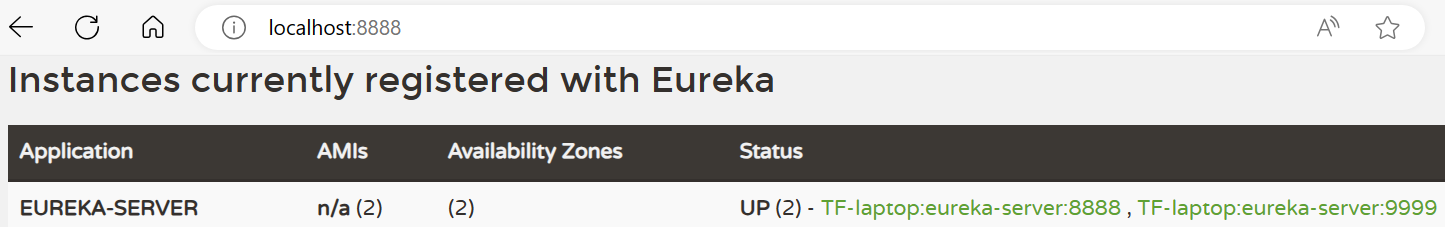
1. 重启eureka-server微服务，然后参考以下内容修改application.yml配置文件







1. 再次启动eureka-server微服务，此时观察并记录eureka中是否注册8888，9999两个eureka-server。



注：若再次启动eureka-server微服务不成功，应设置IDEA允许启动同名服务

