

首先，这还是个Spring boot工程，我们来看下pom。

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>1.5.10.RELEASE</version>
  </parent>

  <groupId>com.xyz</groupId>
  <artifactId>eureka-consumer</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>eureka-consumer</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <jdk.version>1.8</jdk.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-starter-eureka</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
  </dependencies>

  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>org.springframework.cloud</groupId>
        <artifactId>spring-cloud-dependencies</artifactId>
        <version>Dalston.SR1</version>
        <type>pom</type>
        <scope>import</scope>
      </dependency>
    </dependencies>
  </dependencyManagement>

  <build>
```

```

        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.3</version>
                <configuration>
                    <!-- 指定source和target的版本 -->
                    <source>${jdk.version}</source>
                    <target>${jdk.version}</target>
                </configuration>
            </plugin>
        </plugins>
    </build>
</project>

```

然后看下启动类。

```

@EnableDiscoveryClient
@SpringBootApplication
@ComponentScan("com.xyz.controller")
public class Application {

    @Bean
    public RestTemplate restTemplate() {
        return new RestTemplate();
    }

    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }
}

```

标红框的一块初始化了restTemplate，用来真正发起REST请求。

看一下配置。

```

spring.application.name=eureka-consumer
server.port=1003

eureka.client.serviceUrl.defaultZone=http://localhost:1001/eureka/

```

看一下测试的代码。

```

@RestController
public class DemoController {

    @Autowired
    private LoadBalancerClient loadBalancerClient;

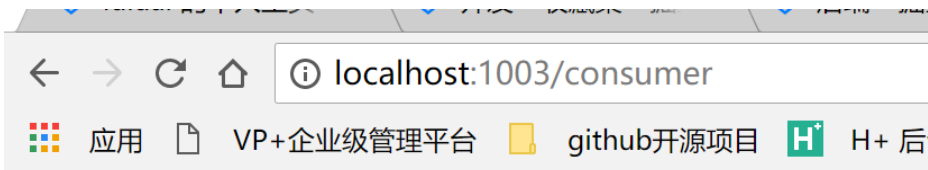
    @Autowired
    private RestTemplate restTemplate;

    @GetMapping("/consumer")
    public String dc() {
        ServiceInstance serviceInstance = loadBalancerClient.choose("eureka-client");
    }
}

```

```
String url = "http://" + serviceInstance.getHost() + ":" + serviceInstance.getPort()
+ "/test";
System.out.println(url);
return restTemplate.getForObject(url, String.class);
}
}
```

首先通过loadBalancerClient接口获取服务提供者的具体实例，然后显示调用提供者这个地址的操作，通过restTemplate发起REST请求并返回结果。



Services: [eureka-client, eureka-consumer]