

homepage 第七期：可用性测试

作者：[ChenforCode](#)

原文链接：<https://ld246.com/article/1573738534048>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



1.简述网关的作用，以course服务为例，

```
@GetMapping("/get/course")
public CourseInfo getCourseInfo(Long id) {
    log.info("<homepage-course>: get course -> {}", id);
    return courseService.getCourseInfo(id);
}
```

如下的服务，如果不通过网关直接访问的话需要的接口地址是

<http://127.0.0.1/homepage-course/get/course?id=10>

就是自己的端口，自己设置的contextPath，但是如果使用网关的话，访问的接口地址会变成

<http://127.0.0.1/chenforcoce/homepage-course/get/course?id=10>

变成的zuul网关的接口，还需要加入zuul配置文件中配置的文件

2.对整个项目进行打包

在项目根目录下输入打包命令 mvn clean package -Dmaven.test.skip=true -U

```
[INFO] imooc-homepage-spring-cloud ..... SUCCESS [ 0.187 s]
[INFO] homepage-eureka ..... SUCCESS [ 2.674 s]
[INFO] homepage-zuul ..... SUCCESS [ 13.526 s]
[INFO] imooc-homepage-service ..... SUCCESS [ 0.002 s]
[INFO] homepage-common ..... SUCCESS [ 3.114 s]
[INFO] homepage-course ..... SUCCESS [ 22.982 s]
[INFO] homepage-user ..... SUCCESS [ 12.287 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 55.015 s
[INFO] Finished at: 2019-11-14T20:40:32+08:00
[INFO] -----
```

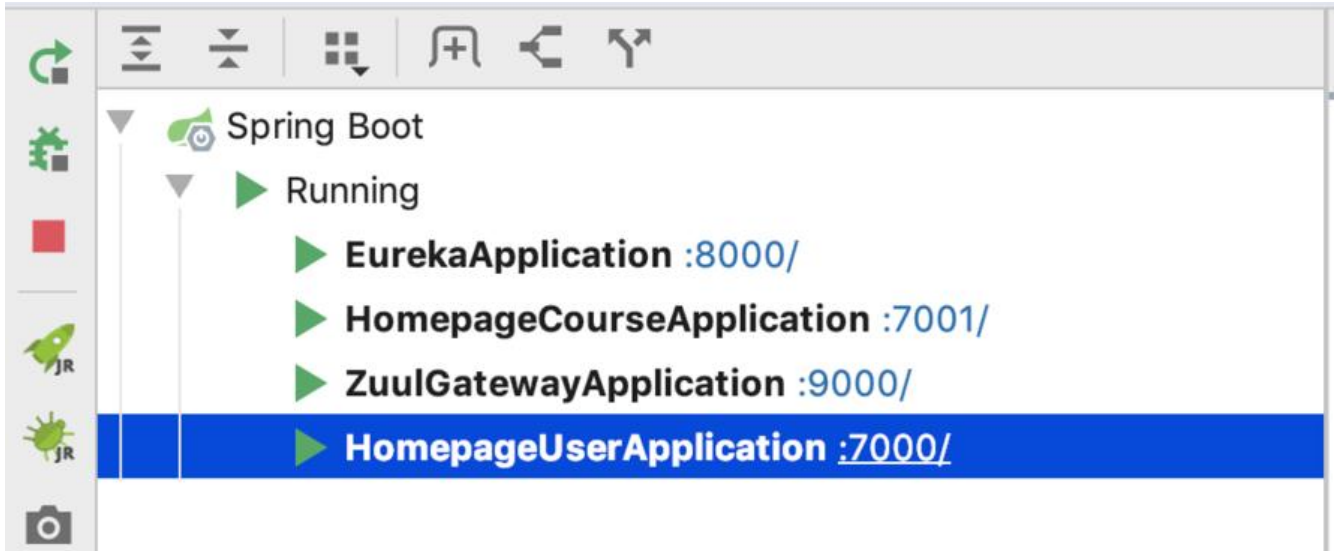
如上图，即为成功

3.之后可以在各自的target目录运行各个服务的jar包

4.另外一种方式就是直接在IDEA中按照顺序启动对应的主函数

尽量按照eureka zuul course user的顺序启动，但是不按照这个顺序也可以，当它们各自稳定下来即

。



在idea中显示四个服务全部启动

5.进入eureka界面，有三个实例

Instances currently registered with Eureka		
Application	AMIs	Availability Zones
EUREKA-CLIENT-HOMEPAGE-COURSE	n/a (1)	(1)
EUREKA-CLIENT-HOMEPAGE-USER	n/a (1)	(1)
HOMEPAGE-ZUUL	n/a (1)	(1)

6.直接用课程服务的接口地址获取课程

GET http://127.0.0.1:7001/homepage-course/get/course?id=8

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

	KEY	VALUE	DESC
<input checked="" type="checkbox"/>	id	8	
	Key	Value	Des

Body Cookies Headers (3) Test Results Status: 200 OK

Pretty Raw Preview Visualize BETA JSON

```
1 {
2   "id": 8,
3   "courseName": "JDK11&12 新特性解读",
4   "courseType": "免费课程",
5   "courseIcon": "http://www.chenforcode.cn",
6   "courseIntro": "解读JDK11&12的新特性"
7 }
```

7.通过网关获取课程

▶ 通过网关获取课程

GET http://127.0.0.1:9000/chenforcode/homepage-course/get/course?id=9

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	id	9
	Key	Value

Body Cookies Headers (3) Test Results Status: 200

Pretty Raw Preview Visualize BETA JSON ↕

```

1  {
2    "id": 9,
3    "courseName": "基于SpringCloud的广告系统的实现",
4    "courseType": "收费课程",
5    "courseIcon": "http://www.chenforcode.cn",
6    "courseIntro": "基于SpringCloud微服务架构的广告系统的实现"
7  }

```

8.其余的测试不再写上去。

9.测试hystrix，即在课程服务不可用的时候，如果访问课程服务，会返回一个空列表。

关闭course服务，然后重新测试访问

GET http://127.0.0.1:9000/chenforcode/homepage-user/get/user/course?id=10

<input checked="" type="checkbox"/>	id	10	
	Key	Value	Description

Body Cookies Headers (3) Test Results Status: 200 OK Time: 30ms

Pretty Raw Preview Visualize BETA JSON ↕

```

1  {
2    "userInfo": {
3      "id": 10,
4      "username": "pkucoder",
5      "email": "pkucoder@qq.com"
6    },
7    "courseInfos": []
8  }

```

这个时候就会返回课程服务中的自定义的hystrix的解决方式，即本地的解决方案

```
/**
 * @author <a href="http://www.chenforcode.cn">PKUCoder</a>
 * @date 2019/11/12 5:12 下午
 * @description 熔断降级策略
 */
@Component
public class CourseClientHystrix implements CourseClient {
    @Override
    public CourseInfo getCourseInfo(Long id) {
        //在出错的时候返回一个无效的courseInfo即可
        return CourseInfo.invalid();
    }

    @Override
    public List<CourseInfo> getCourseInfos(CourseInfosRequest request) {
        //在出错的是否返回一个空列表即可
        return Collections.emptyList();
    }
}
```

10.完毕