资源服务器（即每个业务微服务）的配置，4步

1. 依赖

<**dependency**>  
 <**groupId**>org.springframework.cloud</**groupId**>  
 <**artifactId**>spring-cloud-starter-oauth2</**artifactId**>  
</**dependency**>

1. 加一个这样的类

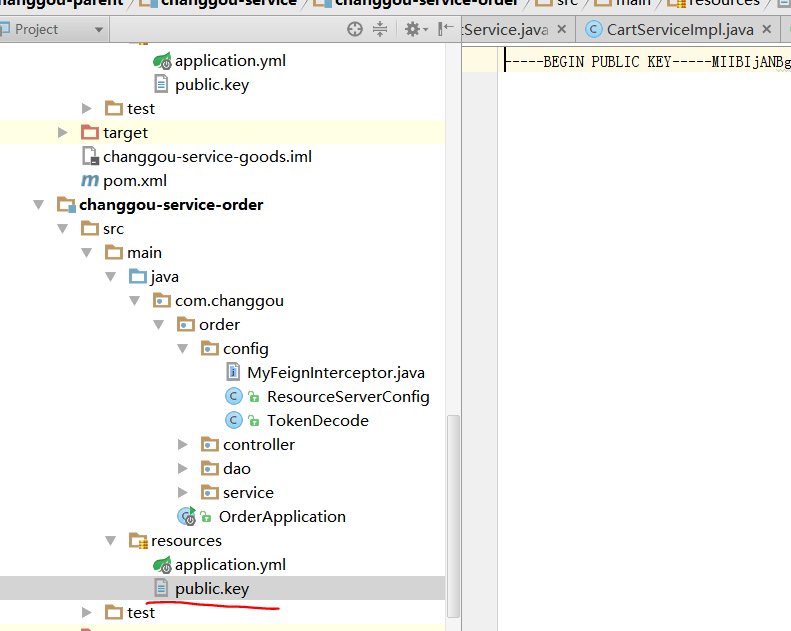
主要功能是用公钥校验token是否正确和http请求哪些路径要校验token



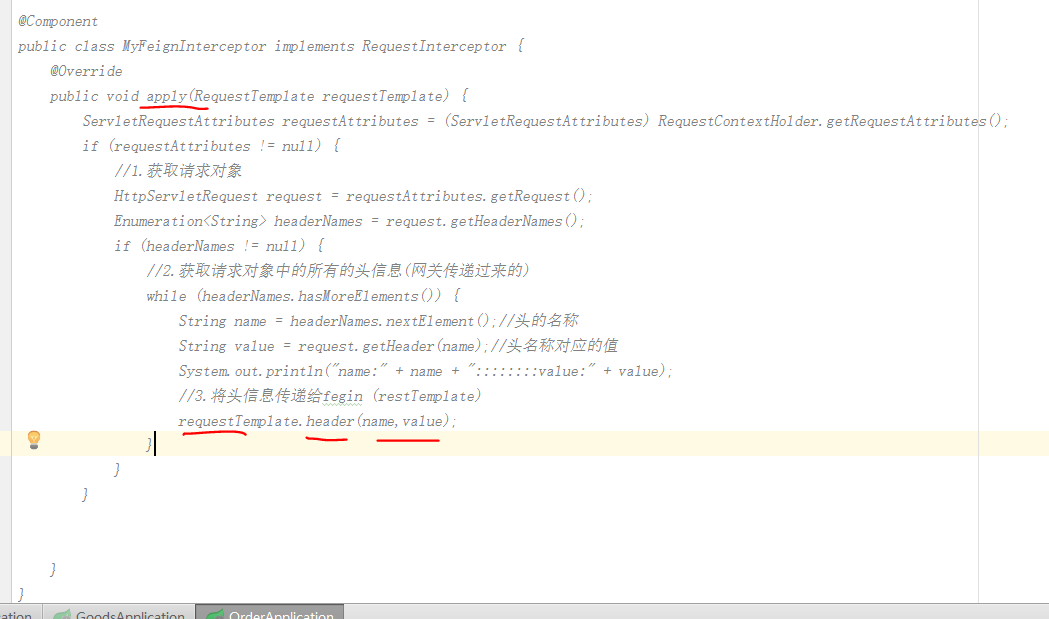




1. 在resource目录下添加公钥



1. Fegin之间的调用，必须把请求头传给下一个微服务

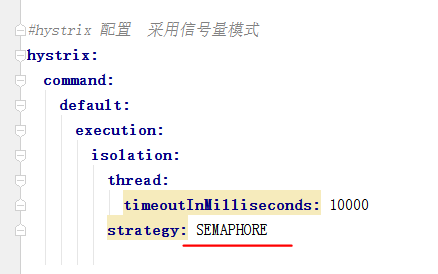


这个方法抽取到common模块里，然后调用方在启动类通过@Bean

注解把它注入到容器中



5.熔断器必须用信号量模式（线程池不行，因为线程池模式的fegin调用会新开一个线程，导致hearder信息无法往下个服务传递）



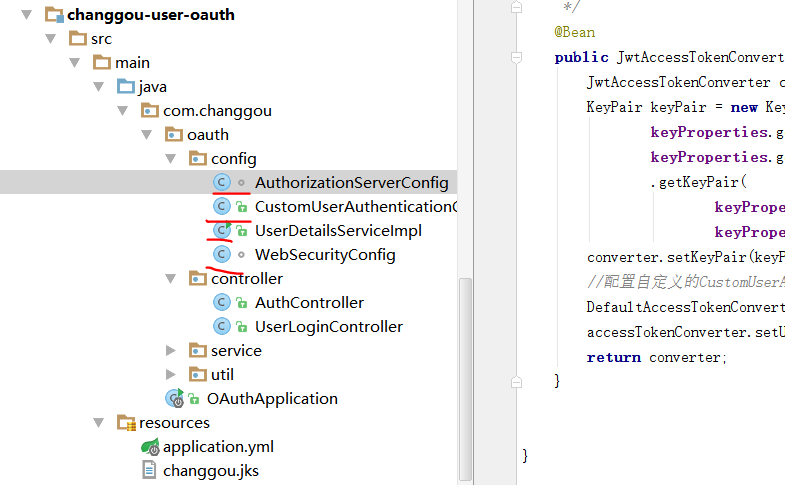
而授权服务需要7步

1. 依赖

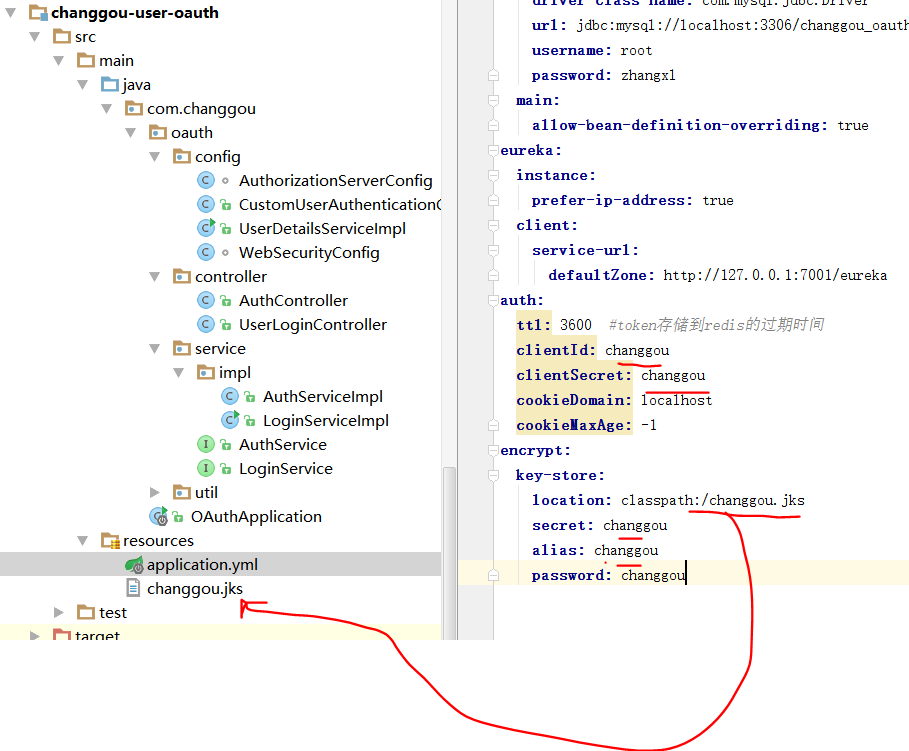
<**dependency**>  
 <**groupId**>org.springframework.security</**groupId**>  
 <**artifactId**>spring-security-data</**artifactId**>  
</**dependency**>  
  
<**dependency**>  
 <**groupId**>org.springframework.cloud</**groupId**>  
 <**artifactId**>spring-cloud-starter-oauth2</**artifactId**>  
</**dependency**>  
  
<**dependency**>  
 <**groupId**>org.springframework.cloud</**groupId**>  
 <**artifactId**>spring-cloud-starter-security</**artifactId**>  
</**dependency**>

1. 添加4个配置类

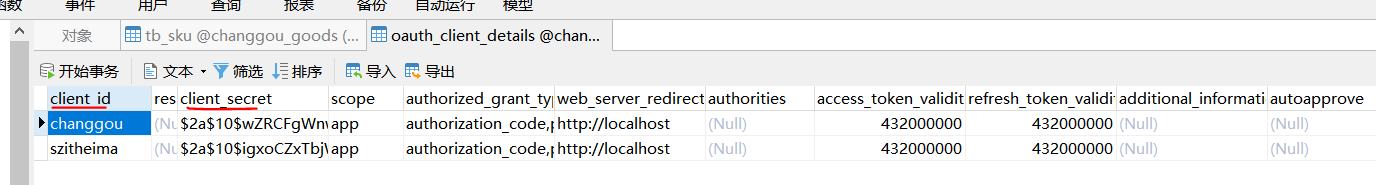




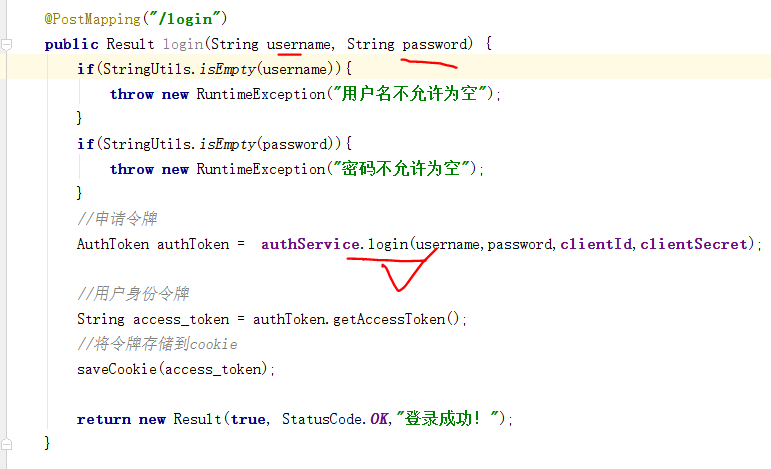
1. 添加属性文件

其中key-store的secret和password是生成证书changou.jks的时候设置的

1. 新建一张表名为oauth\_client\_details的表



1. 模拟前端请求过来调用

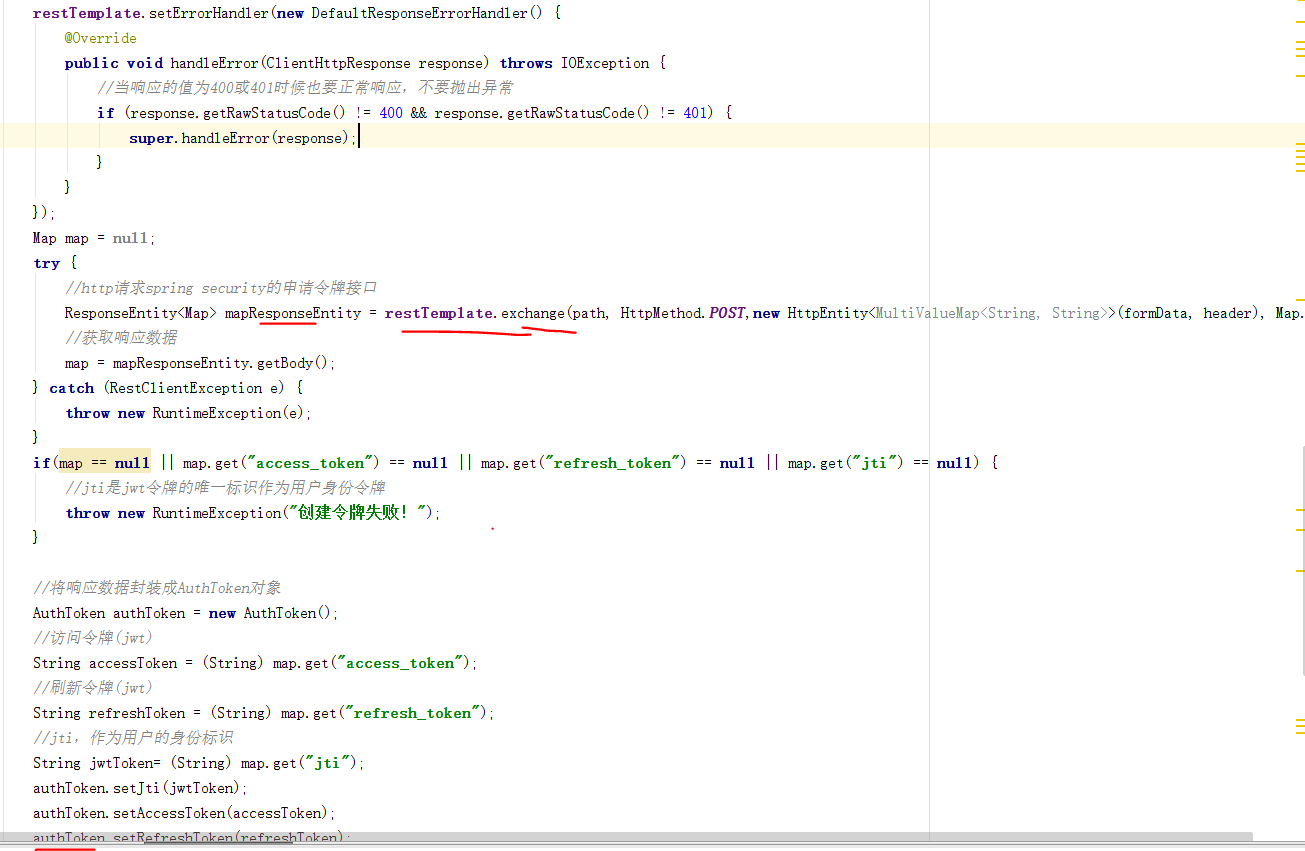


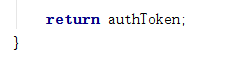
然后我们在后台代码的authServce.login()里去调用/oauth/token地址，这是真正的获取token











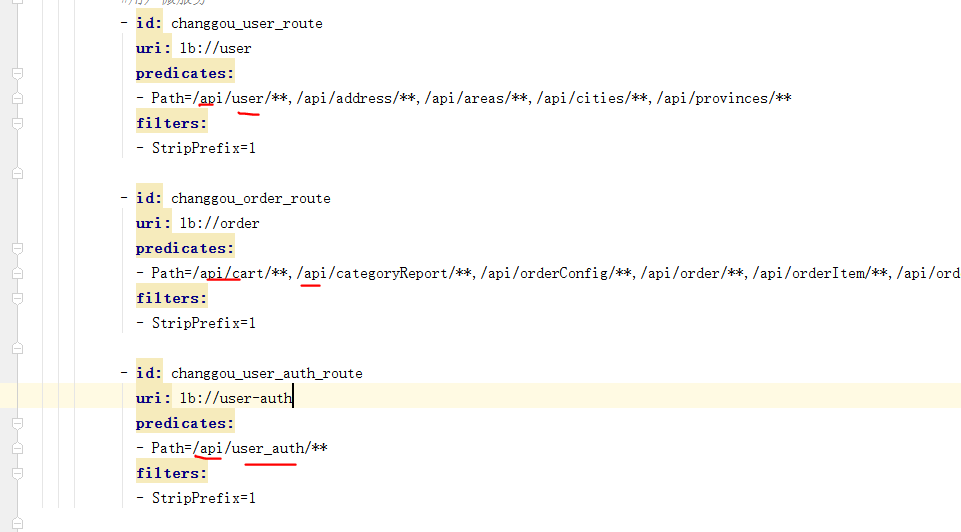
其中restTemplate.exchange(xxxxx),这个时候是实际去调用UserDetailsServiceImpl类的的loadUserByUsername（）方法

网关部分

添加两个类（主要是注册登录的请求不拦截，其他的要校验token）



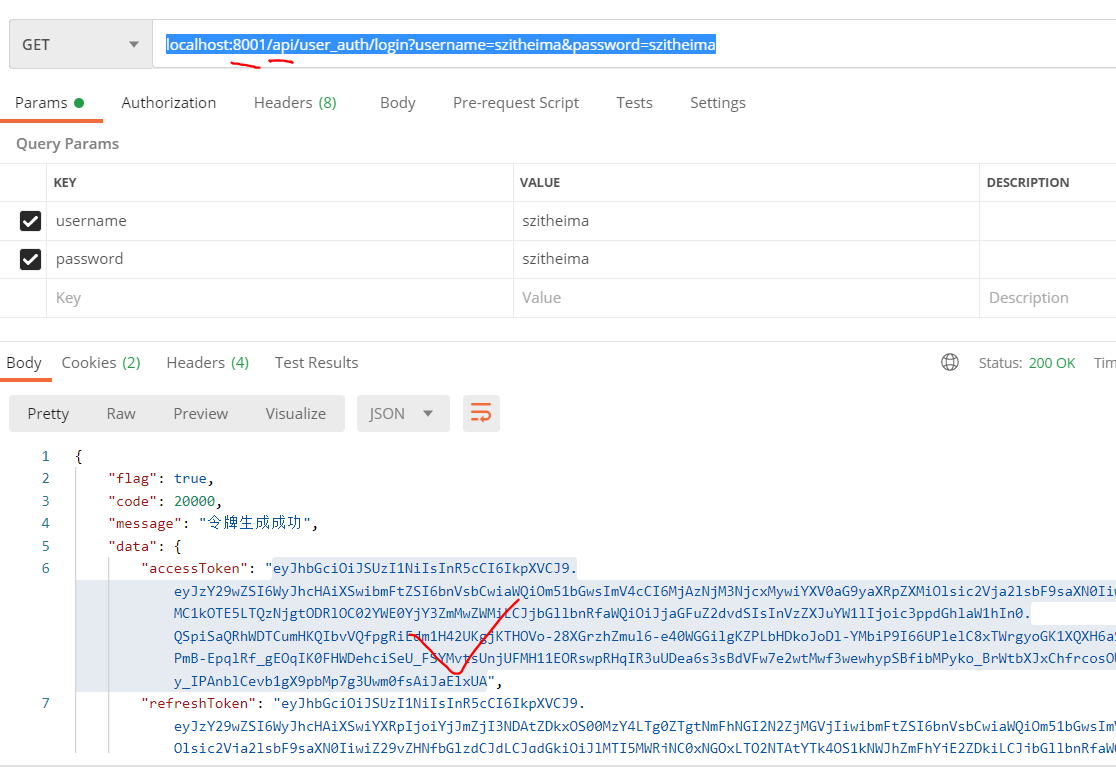
网关的属性配置



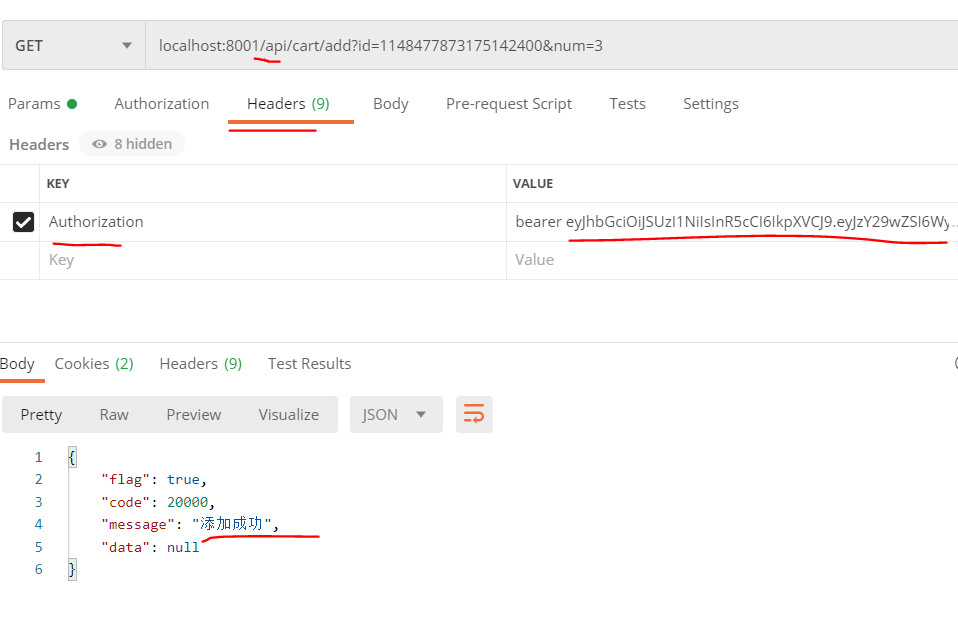
程序运行（启动注册中心，网关，授权服务，用户服务，订单服务，goods服务）

先通过登录获取token

localhost:8001/api/user\_auth/login?username=szitheima&password=szitheima（8001是网关的端口）



然后请求接口的时候带上token（localhost:8001/api/cart/add?id=1148477873175142400&num=3）



短信登录

1.添加一个类SmsCodeTokenGranter自定义授权器类继承AbstractTokenGranter

，重写 getOAuth2Authentication 函数，在这个函数中，自定义授权认证逻辑；

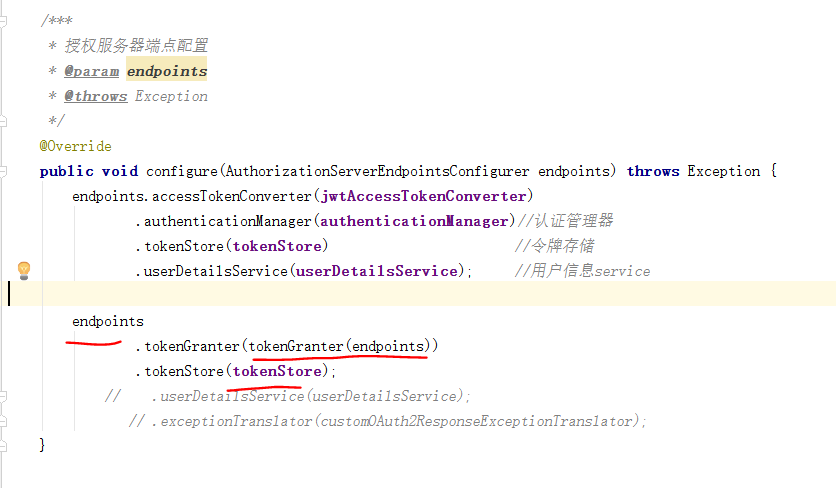
校验手机号 + 短信验证码是否一致，在getOAuth2Authentication， 手机和验证码校验无误， 返回 OAuth2Authentication 授权信息（access\_token）

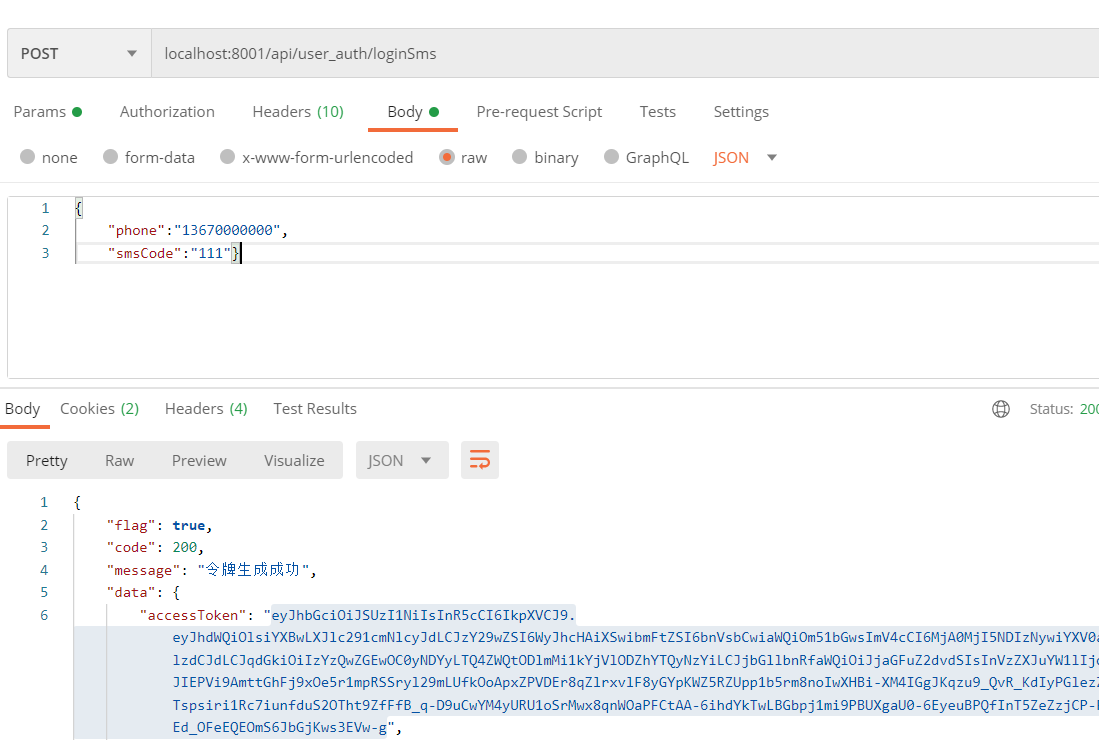
2.在AuthorizationServerConfig类里 做两步：

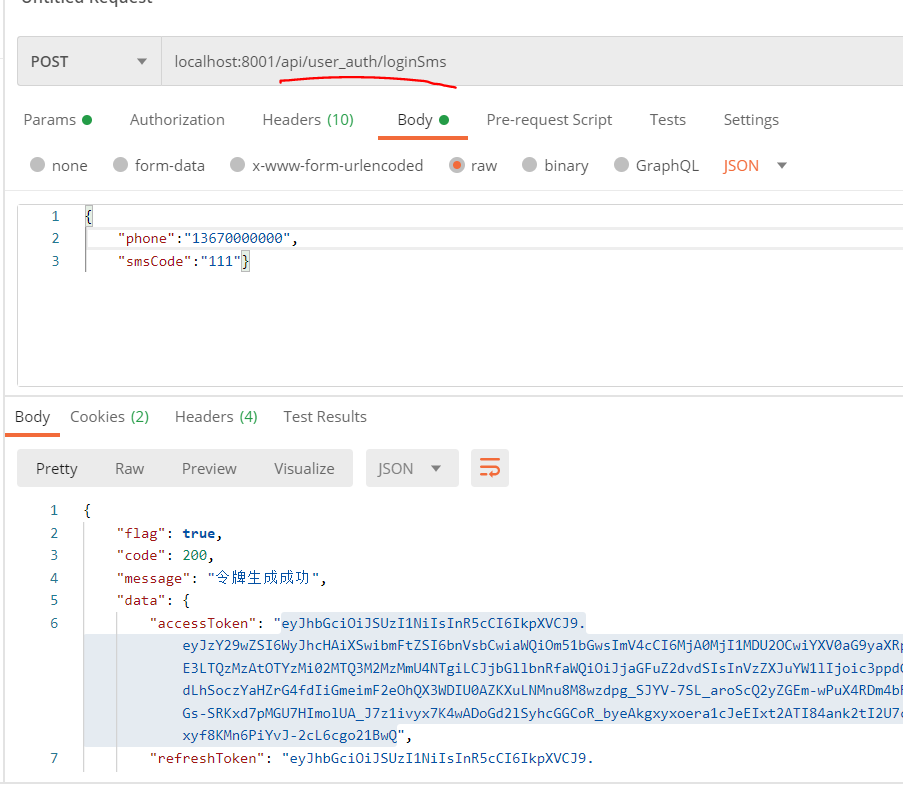
一．添加自定义授权类型



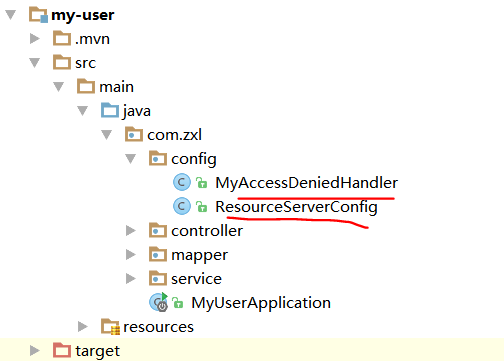
### 二．添加所有端点信息







在资源服务（即业务微服务） 添加resoureid和没有权限访问的返回handler



1. 添加一个MyAccessDeniedHandler类
2. 在资源配置类ResourceServerConfig类里配置（resourceId）和MyAccessDeniedHandler

