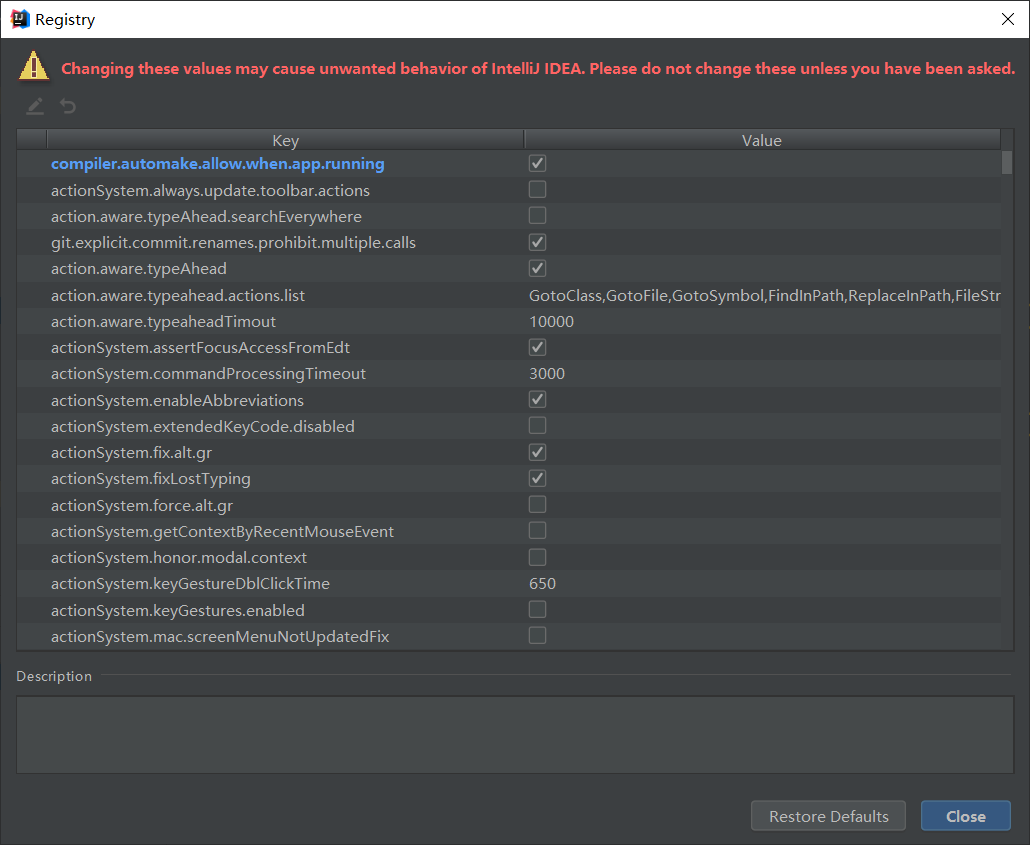
@RestController写成@Controller

80端口写成@PostMapping没有改成@GetMapping



热部署选running，该选项不一定是第一个，可能要往下找

@RequestBody写成@ResponseBody，消费端不需要写@RequestBody，服务提供端需要写

项目名cloud-api-commons写成commons-api-commons，更改过来后父工程pom中要修改<module> common-api-commons</module>为<module>cloud-api-commons </module>

eureka:  
 instance:  
 hostname: localhost *#eureka服务端的实例名字* client:  
 register-with-eureka: false *#表识不向注册中心注册自己* fetch-registry: false *#表示自己就是注册中心，职责是维护服务实例，并不需要去检索服务* service-url:  
 defaultZone: http://${eureka.instance.hostname}:${server.port}/eureka/ *#设置与eureka server交互的地址查询服务和注册服务都需要依赖这个地址*

service-url:存在缩进问题，要和fetch-registry对齐

defaultZone: http://eureka7001.com:7001/eureka,http://eureka7002.com:7002/eureka

中间是逗号隔开

spring-cloud-netflix-eureka-client自带spring-cloud-starter-ribbon引用，无需重复引入

@FeignClient(value = "CLOUD-PAYMENT-SERVICE")

@FeignClient要后面的value指定对应的服务名

服务端要用@GetMapping而不是@PostMapping

服务端的public CommonResult create(Payment payment)是没有@RequestBody的

@FeignClient(value = "CLOUD-PROVIDER-HYSTRIX-PAYMENT")缺少(value = "CLOUD-PROVIDER-HYSTRIX-PAYMENT")

消费端@HystrixCommand放在controller的方法上

生产端@HystrixCommand放在service的方法上

消费端.yml文件要加：

feign:  
 hystrix:  
 enabled: true

只添加了以下依赖

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-netflix-hystrix-dashboard</artifactId>  
</dependency>

@EnableHystrixDashboard不生效

要添加以下两个依赖：

<dependency>  
 <groupId>com.netflix.hystrix</groupId>  
 <artifactId>hystrix-javanica</artifactId>  
 <version>RELEASE</version>  
</dependency>  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-netflix-hystrix-dashboard</artifactId>  
</dependency>

原因：idea问题没有自动导入hystrix项目中的jar依赖，两个依赖导入完成后可删除第一个依赖

因国内访问GitHub速度慢，访问<http://config-3344.com:3344/master/config-dev.yml>时耐心等待，不是错误只是访问速度慢

server:  
 port: 3344  
spring:  
 application:  
 name: cloud-config-center  
 cloud:  
 config:  
 server:  
 git:  
 uri: https://github.com/gddjm/springcloud-config.git  
 search-paths:  
 - springcloud-config  
 label: master  
 rabbitmq:  
 host: localhost  
 port: 5672  
 username: guest  
 password: guest  
  
eureka:  
 client:  
 service-url:  
 defaultZone: http://localhost:7001/eureka  
  
management:  
 endpoints:  
 web:  
 exposure:  
 include: 'bus-refresh'

注意rabbitmq的缩进

**问题：Nacos1.2.0 数据持久化 mysql，不支持mysql8及其以上版本**

**解决方法：**[**https://www.cnblogs.com/gyli20170901/p/11245270.html**](https://www.cnblogs.com/gyli20170901/p/11245270.html)

在Mac/Linux中执行.sh脚本，异常/bin/sh^M: bad interpreter: No such file or directory。  
分析：这是不同系统编码格式引起的：在windows系统中编辑的.sh文件可能有不可见字符，所以在Mac/Linux系统下执行会报以上异常信息。

方法1：在NotePad++中

编辑->文档格式转换->转为Unix。

方法2：

也可在Mac/Linux中转换：

首先要确保文件有可执行权限

#sh>chmod a+x yourFileName

然后修改文件格式

#sh>vi yourFileName

利用如下命令查看文件格式

:set ff 或 :set fileformat

可以看到如下信息

fileformat=dos 或 fileformat=unix

利用如下命令修改文件格式

:set ff=unix 或 :set fileformat=unix

保存后退出  
最后再执行文件

#sh>./filename

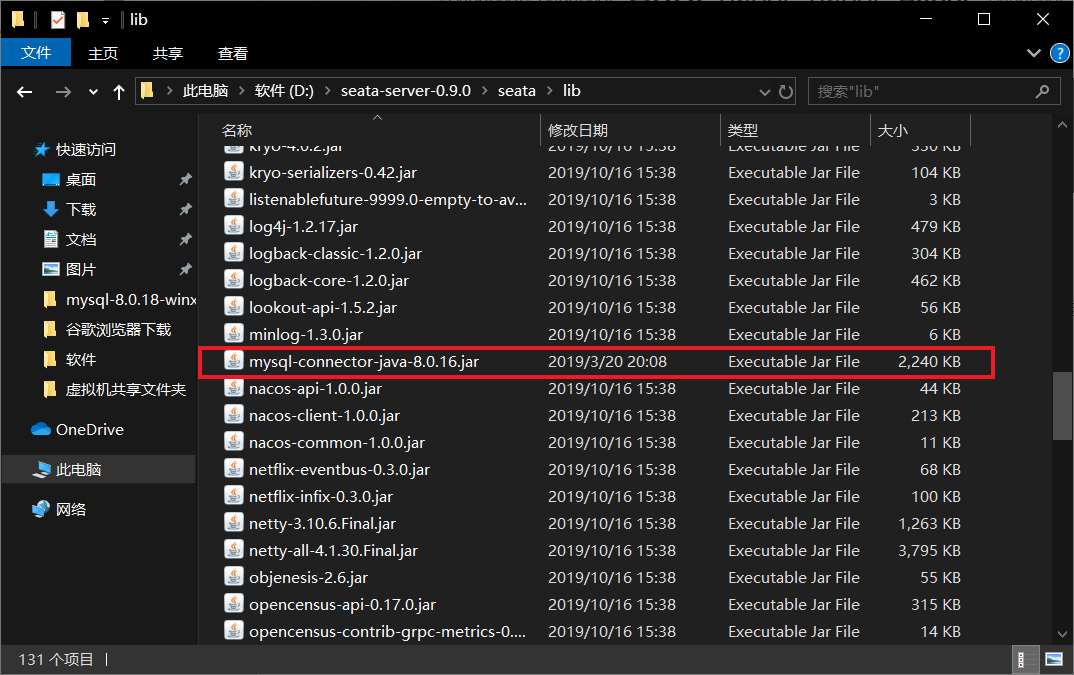
@GetMapping("/testHotKey")  
@SentinelResource(value = "testHotKey", blockHandler = "deal\_testHotKey")  
public String testHotKey(@RequestParam(value = "p1",required = false)String p1,  
 @RequestParam(value = "p2",required = false)String p2){  
 *log*.info("testHotKey测试热点规则");  
 return "------testHotKey";  
}  
  
public String deal\_testHotKey(String p1, String p2, BlockException exception){  
 return "-----deal\_testHotKey,o(╥﹏╥)o";  
}

deal\_testHotKey方法忘记写BlockException exception参数

问题：seata不支持mysql8及以上版本

解决方法：

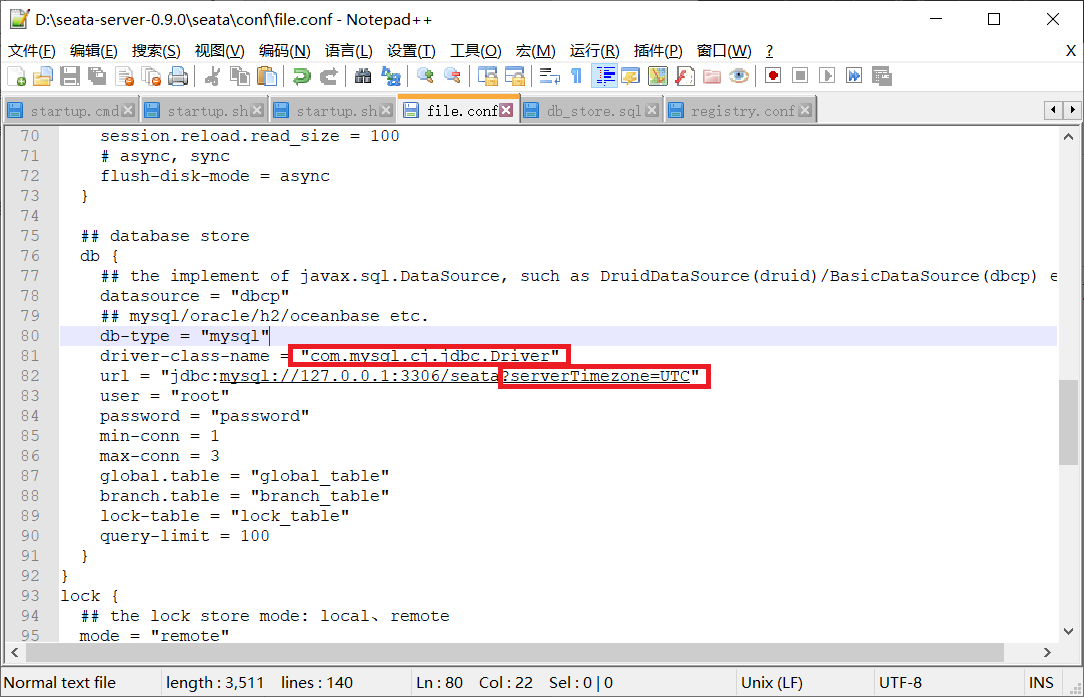
1. 用mysql8版本的连接驱动替换掉旧版本的连接驱动。去到seata/lib目录下，将8版本的jar包粘贴到目录下



1. 修改seata/conf/file.conf文件，内容：

driver-class-name = "com.mysql.cj.jdbc.Driver"

url = "jdbc:mysql://127.0.0.1:3306/seata?serverTimezone=UTC"



问题：Idea的pom.xml报错： Failed to read artifact descriptor for XXX

原因：网速不好，下载依赖包出错，但idea认为依赖包已下载完成，所以不再下载，但springboot应用找不到jar包，报错。

删除repository中对应的jar包文件夹，maven reimport，即重新让maven下载依赖包就好了。

Seata的bin目录下的file.conf和放在项目中的file.conf内容有所区别：

1）bin目录下的file.conf：

service {  
 #vgroup->rgroup  
 **vgroup\_mapping.my\_test\_tx\_group = "fsp\_tx\_group"**  
 #only support single node  
 default.grouplist = "127.0.0.1:8091"  
 #degrade current not support  
 enableDegrade = false  
 #disable  
 disable = false  
 #unit ms,s,m,h,d represents milliseconds, seconds, minutes, hours, days, default permanent  
 max.commit.retry.timeout = "-1"  
 max.rollback.retry.timeout = "-1"  
}

2）放在项目中的file.conf:

service {  
 #vgroup->rgroup  
 **vgroup\_mapping.fsp\_tx\_group = "default"**  
 #only support single node  
 default.grouplist = "127.0.0.1:8091"  
 #degrade current not support  
 enableDegrade = false  
 #disable  
 disable = false  
 #unit ms,s,m,h,d represents milliseconds, seconds, minutes, hours, days, default permanent  
 max.commit.retry.timeout = "-1"  
 max.rollback.retry.timeout = "-1"  
}