**一，个别店员查不到账号**

店员账号1002查不到账号，因为temp\_data\_scope.erp\_store\_id IN ( '' ) ，没有erp\_store\_id，没有获取ERP系统的id。OrderMapper.xml里的 selectPage1查询。

|  |
| --- |
| OrderInfoServiceImpl.java  public IPage<OrderInfo> page1(IPage<OrderInfo> page, Wrapper<OrderInfo> queryWrapper) {  IPage<OrderInfo> orderInfoIPage = baseMapper.selectPage1(page, queryWrapper.getEntity()); //这里没有获取到erp\_store\_id |

10223有，对比后台日志的sql语句得出以上结论。

**二，项目信息，数据库信息**

1. **相关表**

订单表：order\_info，有用户信息字段：user\_id

订单子表：order\_item

对应关系order\_info : id = order\_item : order\_id

购物车表：shopping\_cart

sys\_store ：门店表

home\_goods\_category ： 首页分类库

goods\_label ： 首页活动，首页标签库

user\_info ：会员表

user\_code\_info：包含用户的推荐人，推荐门店信息的表（user\_id对应user\_info表的主键id）

notice\_item: 首页轮播图，及首页通知消息表

1. **订单详情接口**

OrderInfoController.java 🡪 @GetMapping("/{id}") (117行)

1. **小程序，个人订单接口**

UserInfoApi.java 🡪 getUserOrderInfo(..)

1. **积分所操作在接口: OrderInfoApi.java**

**三，访问接口**

1，首页模块访问接口

ActivityInfoApi.java

|  |
| --- |
| @ApiOperation(value = "首页模块列表")  @GetMapping("/activity/list")  public R activityList() { |

2，查看个人订单接口

OrderInfoApi.java

|  |
| --- |
| @GetMapping("/page")  public R getOrderInfoPage(Page page, OrderInfo orderInfo) { } |

3，购物车商品接口 ShoppingCartApi.java

|  |
| --- |
| @GetMapping("/page")  public R getShoppingCartPage(Page<ShoppingCart> page, ShoppingCart shoppingCart) { } |

更改购物车内商品数量

|  |
| --- |
| public R edit(@RequestBody ShoppingCart shoppingCart){ } |

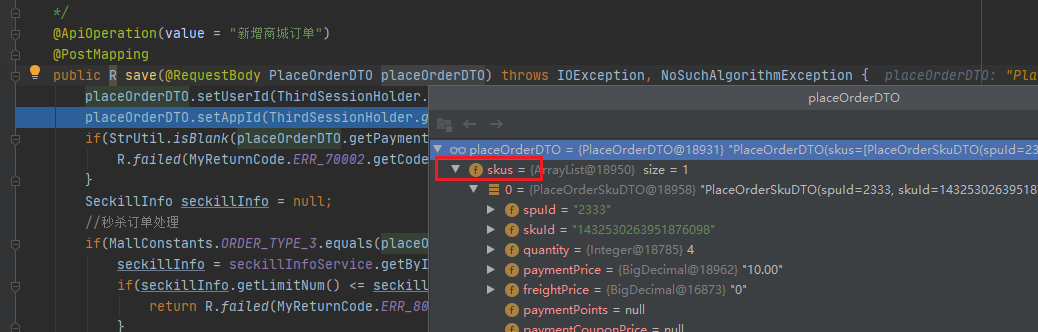
4，保存订单相关操作接口。

1. 保存订单流程，OrderInfoApi.java

public R save(.. ){}

购物车内的数据在skus属性里

购物车总价字段是salesPrice，是计算出来的

(2)

OrderInfoServiceImpl{

420： orderItem.setSalesPrice(placeOrderSkuDTO.getPaymentPrice()); //设置价格

}

5，导入导出表格接口

ExportExcelController.java

**四，其它**

1. **后台查询所有商品报错与promotionId有关**，**sql : GoodsSpuMapper.xml**

(1) OrderInfoServiceImpl.java

|  |
| --- |
| importGoodsPromotion(..){ } |

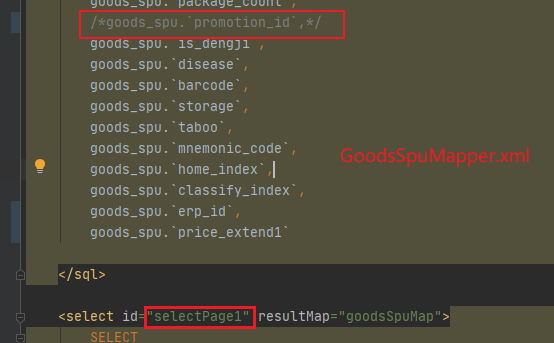
<select id="selectPage1" resultMap="goodsSpuMap">

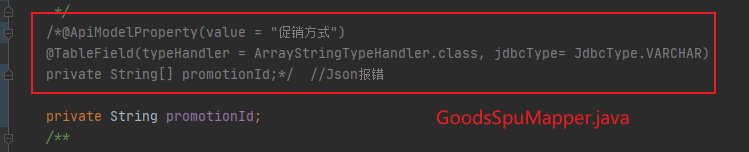
(GoodsSpuServiceImpl.java:83)

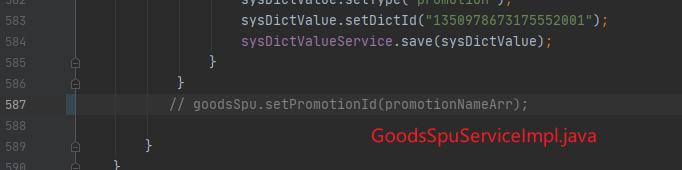
|  |
| --- |
| 异常信息：  nested exception is org.apache.ibatis.executor.result.ResultMapException: Error attempting to get column 'promotion\_id' from result set |

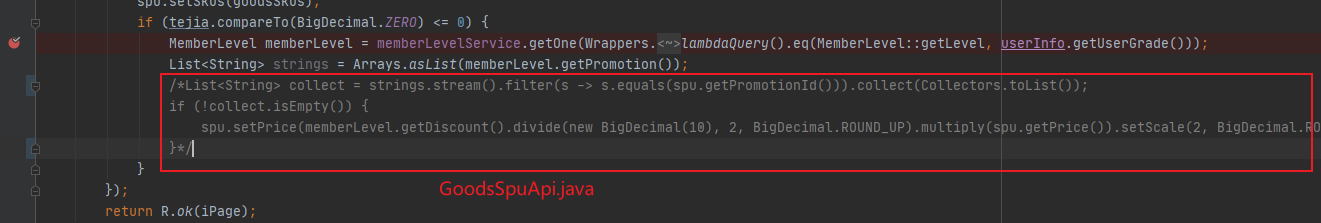
本地运行小程序，访问首页商品相关也报此错

**先注释，不报错，个人注释之处**：









protmotionId用处：

OrderInfoServiceImpl.java{

357: List<String> strings = Arrays.asList(memberLevel.getPromotion());

427-431: List<String> collect = strings.stream().filter。。。。。

}

**2, 租户，tenanat\_id没有用**

**3, 个人id: 1523851647128715265 user\_info表**

**五，后台小程序首页轮播图上传报错**

**重启base-upms-admin模块恢复初始化设置即可**

**六，区域价格，美团判断配送范围，后台指定美团配送时**

* 客户小程序下单后，指定给客户附近门店，优先选择美团配送，如超范围美团无法送，则订单划拨到总店，发快递。
* 美团数据库存有天宇门店信息，包括位置，名称等数据(门店表sys\_store，erp\_id就是对应美团数据的门店id)。
* 客户下单地址与他定位地址不同时，要根据下单地址确定价格，并给出提示

相关接口

(1)

MeiTuanOpenController.java

createOrder(){}

MeiTuanOpenServiceImpl.java

createOrder() : 73行，抛出超范围异常：送件地址超出当前可配送范围

checkOrder(){} :方法，校验配送能力

(2)

小程序确认订单页面，去预约（确认购买，美团判断流程）

接口：mall/api/ma/orderinfo，默认走save方法，保存订单

OrderInfoApi.java 🡪

public R save{

(137行) OrderInfo orderInfo = orderInfoService.orderSub(placeOrderDTO,seckillInfo); //调用方法判断美团是否配送

}

(3)

OrderInfoServiceImpl.java

524行判断订单是否能由美团配送

|  |
| --- |
| JSONObject meiTuanObject=meiTuanOpenService.checkOrder(orderInfo.getStoreId(), userAddress.getLocation(),userAddress.getAddressName()+userAddress.getDetailInfo()); |

(4)小程序下单页面，门店列表

StoreAPI.java

数据库查询语句，根据用户坐标判断附近门店，标红部分

|  |
| --- |
| SELECT  \*,  **( st\_distance ( point ( longitude, latitude ), point ( 114.540352, 37.085315 )) / 0.0111 ) AS distance**  FROM  sys\_store  WHERE  ENABLE = 1  AND longitude IS NOT NULL  AND latitude IS NOT NULL  AND ( date\_format( now(), '%T' ) BETWEEN business\_start\_time AND business\_end\_time )  AND sys\_store.tenant\_id = '1'  ORDER BY  distance ASC |

**七，数据库表及配置文件修改 ，发布项目时注意恢复或更新**

新增：

goods\_spu\_area\_price

goods\_package

goods\_package\_spu

修改：**注意修改表不要直接导入，会覆盖正式数据!!!!**

shopping\_cart ：增加字段package\_id, package\_name, remark

sys\_store ： 增加区域价格标记字段in\_city，area\_price\_column，

**配置文件修改**

base-mall-admin-dev.yml

mall:

#notify-host: http://ty.ukims.cn #正式环境需恢复

notify-host: http://localhost:8082

**八，GoodsSku的保存时机**

在保存或编辑GoodsSpu的同时保存或修改了spu对应的skus。

套餐保存时也应保存对应的sku信息，否则下单时报错。

**九，套餐付款后，微信回调步骤**

未付款状态，无法退款，原因：

1，生成付款二维码接口：

OrderInfoApi.java

/unifiedOrder

使用feignWxPayService.unifiedOrder()调用微信模块，微信模块再请求微信官方支付接口，返回支付二维码等数据给前台

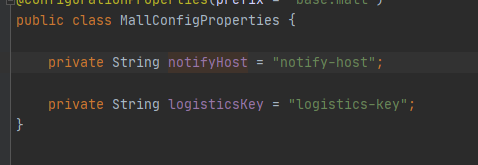
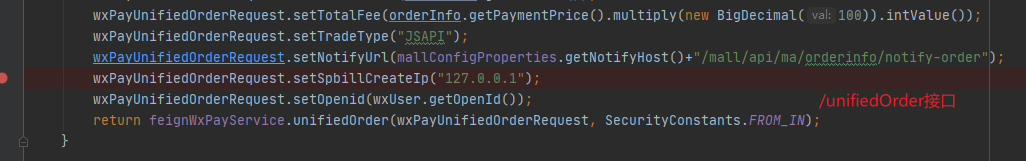
1. 支付后调用的接口，在下单接口传项目的回调接口uri，确认支付成功后，微信官方服务器会回调本项目的接口（**不是本地服务自己调用的**）， 接口内进行更新订单信息的支付状态is\_pay，订单状态status等字段（表：order\_info）

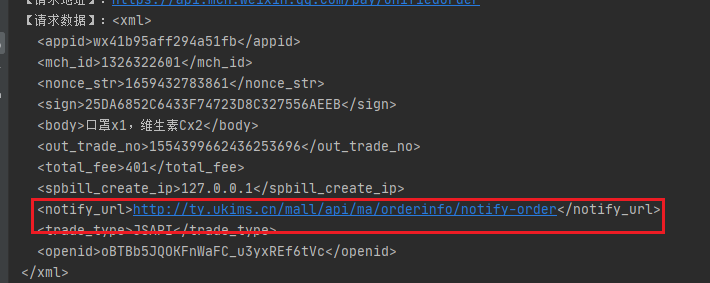
/notify-order

注意：这个接口是微信服务器回调的，前端小程序没有请求接口

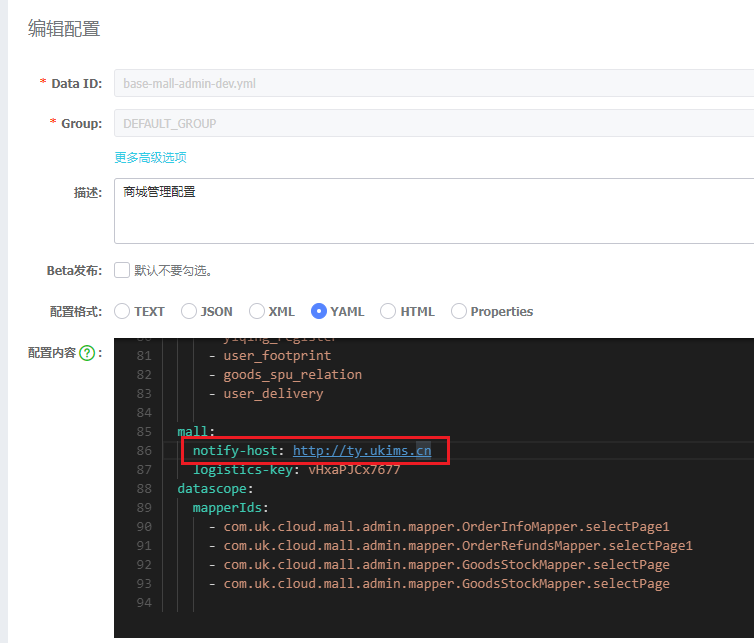
**参考：https://blog.csdn.net/qq\_38669394/article/details/106667053**

**原因：支付后本地服务回调了线上的回调接口更新了线上数据库，但线上数据库没有订单信息，因此也无法更新，本地的数据也没更新**

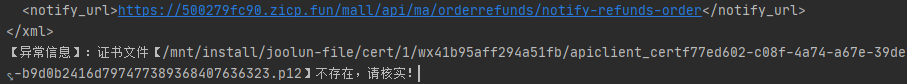




在Nacos配置中心修改地址为本地即可



3，本地测试退款报错，正数文件不存在，需要本地配置证书



证书配置在后台小程序管理页面

微信支付，退款正式环境配置

数据库：base\_wx-->表wx\_app--列key\_path