Git ：

c语言 分布式版本控制 不必联网 分支管理

git init 初始化一个 git 仓库

git add <file> 文件添加到仓库

git commit 提交至仓库、 -m 输入本次提交说明，任意内容

git status 查看被修改的文件

git diff 查看 修改后的内容

git log 显示由近到远的提交日志 --pretty=oneline参数

HEAD表示当前版本 上个版本HEAD^ 上上个HEAD^^ Head~100

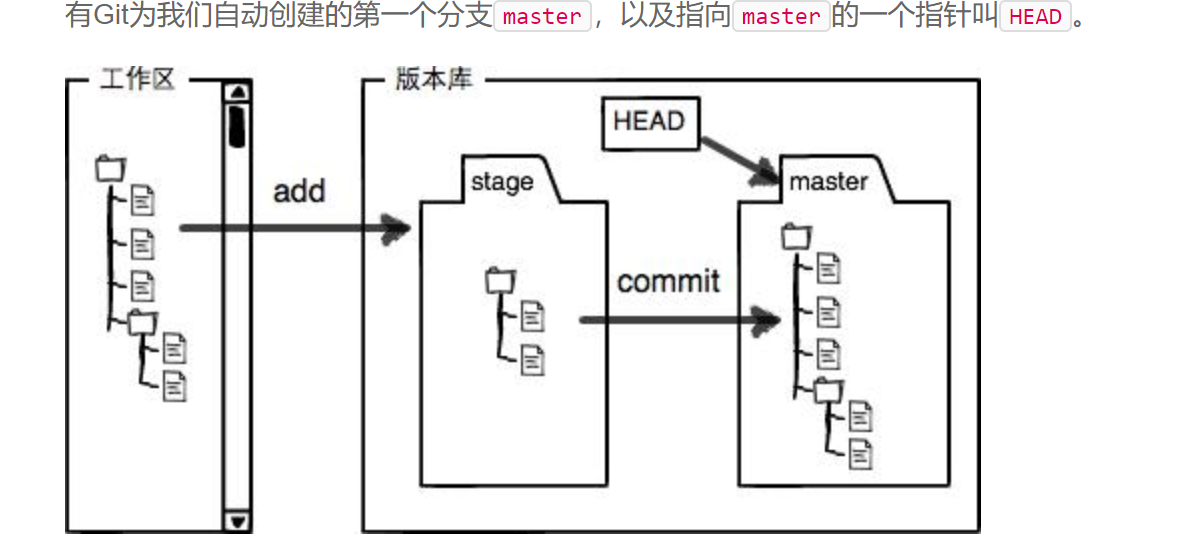
回退上版 git reset --hard HEAD^

git reflog 用于记录每一次命令，可查看命令历史

工作区（Working Directory）： 本地文件目录

版本库（Repository）： .git 目录

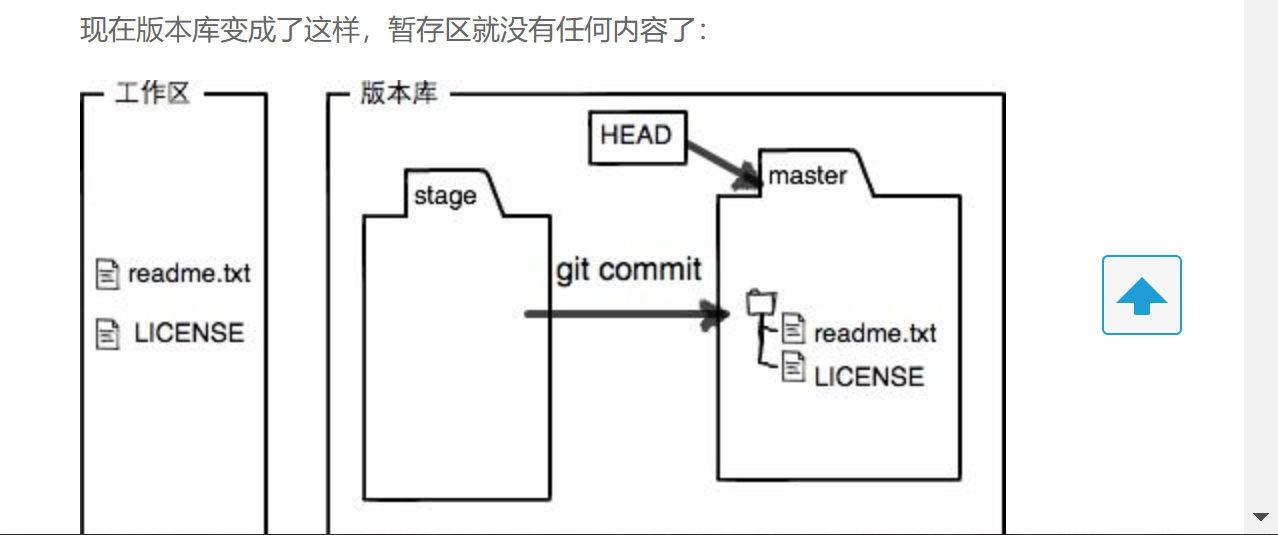
包括 stage （index）暂存区，第一个分支master 以及指向master的指针HEAD



git add把文件添加进去，实际上就是把文件修改添加到暂存区；

用git commit提交更改，实际上就是把暂存区的所有内容提交到当前分支。

提交后



Git的跟踪修改：每次修改，如果不用git add到暂存区，那就不会加入到commit中。

git checkout -- file 撤销修改（如无-- 则切换分支）

命令git reset HEAD <file>可以把暂存区的修改撤销掉（unstage），重新放回工作区

git reset命令既可以回退版本，也可以把暂存区的修改回退到工作区。当我们用HEAD时，表示最新的版本。

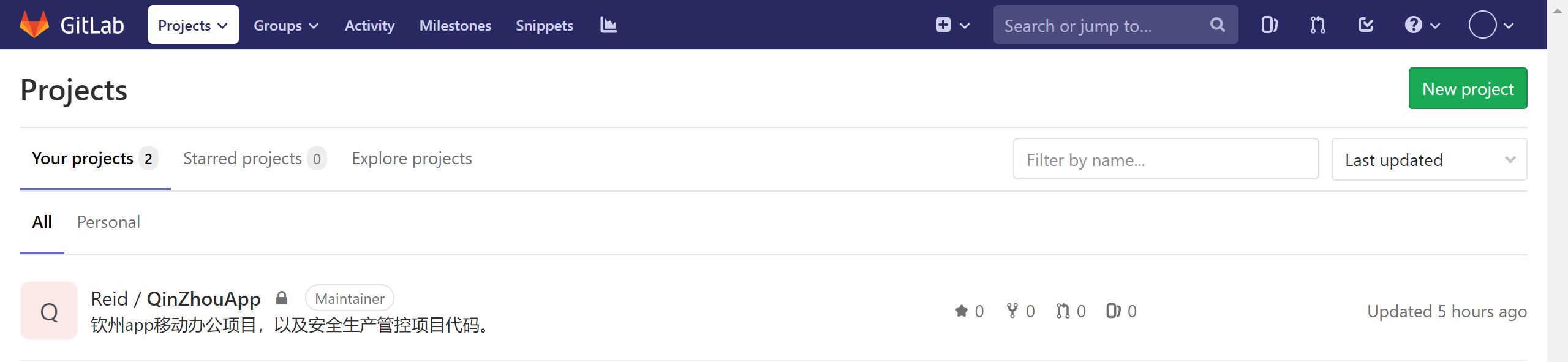
git rm filename 删除文件 git checkout --filename 撤销删除

公司代码git仓库地址： <http://git.sep.i1e.cc:9205/> 学习时新建自己的项目。

钦州app导入

Git 地址：<http://git.sep.i1e.cc:9205/admin/users/liyuting>

有权限才能看见项目



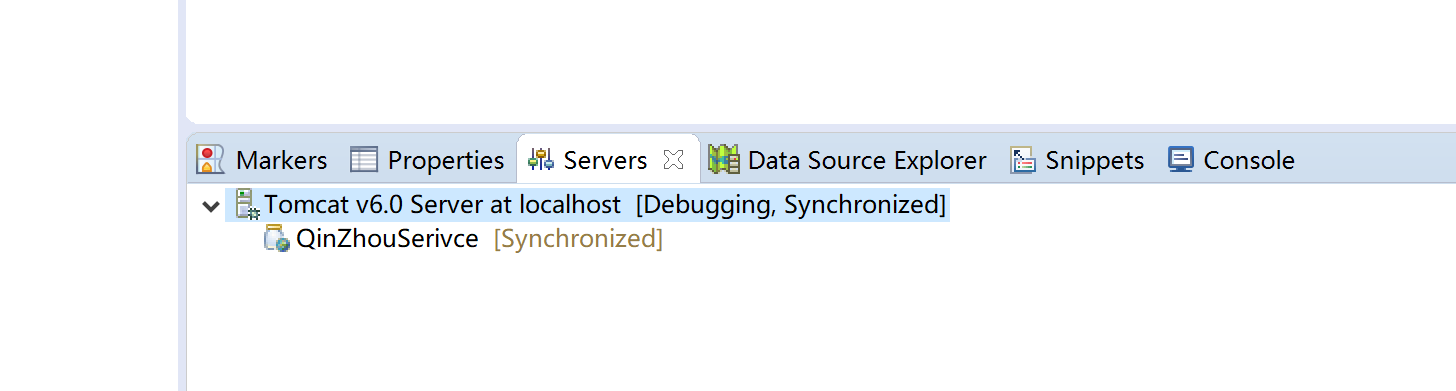
账密：

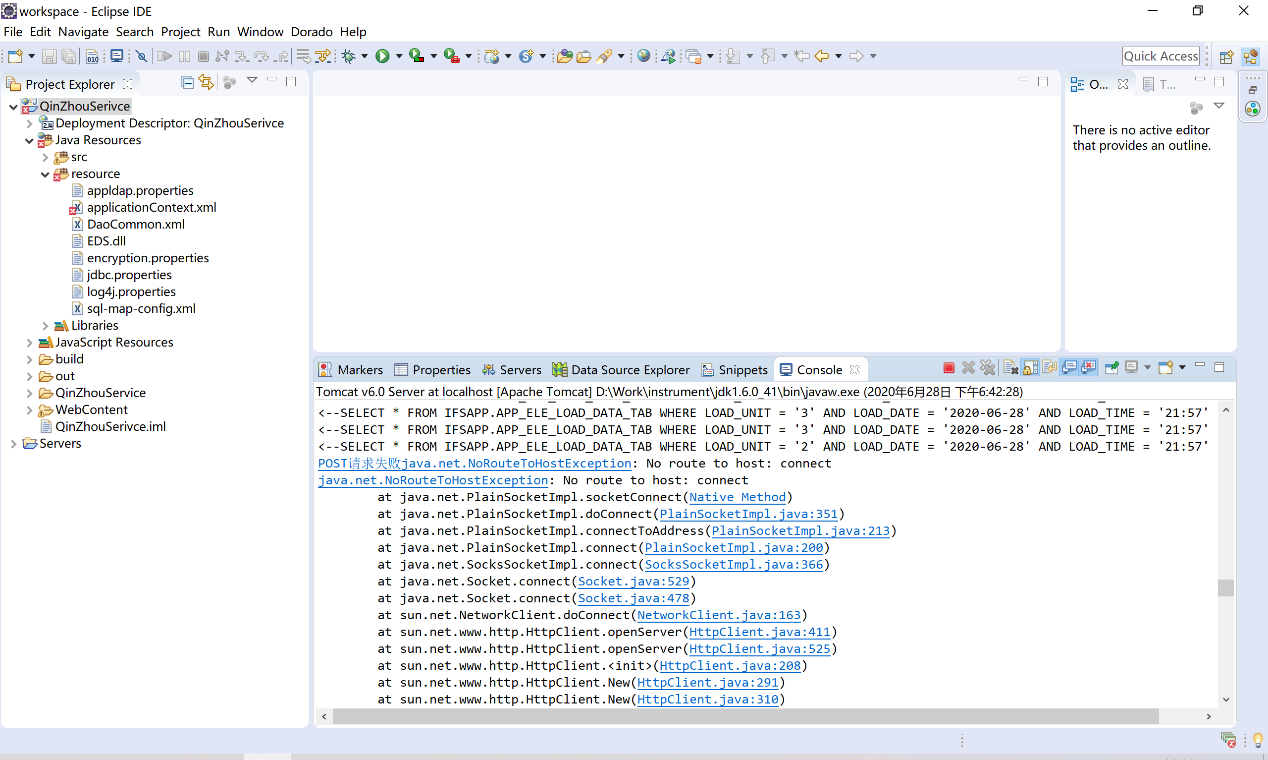
下载项目zip解压 路径：D:\Work\Qingzhou\qinzhouapp-master

eclipse导入文件夹，打开项目QinzhouService tomcat选择6.0 jdk1.6 绿色版无需安装

绿色版文件路径：D:\Work\instrument

将项目添加到tomcat运行





无vpn会post请求不到

安装nvm（如装过nodejs，卸载。删除npm相关环境变量，删除相关文件夹，AppDate是app缓存等文件，进去找npm相关删，别全删！！！！）

安装完成后 cmd

nvm看看安对了没

nvm install 项目的nodejs版本（8.9.3）

安装成功后 nvm use 8.9.3

npm 命令进入npm安装

npm install -g ionic@3.9.2（注意不要升版本 升了uninstall）

解决报要装python的错

npm install --global node-gyp

node-gyp，是由于node程序中需要调用一些其他语言编写的 工具 甚至是dll，需要先编译一下，否则就会有跨平台的问题，例如在windows上运行的软件copy到mac上就不能用了，但是如果源码支持，编译一下，在mac上还是可以用的。node-gyp在较新的Node版本中都是自带的（平台相关），用来编译原生C++模块。



npm install -g cordova@8

安装cordova

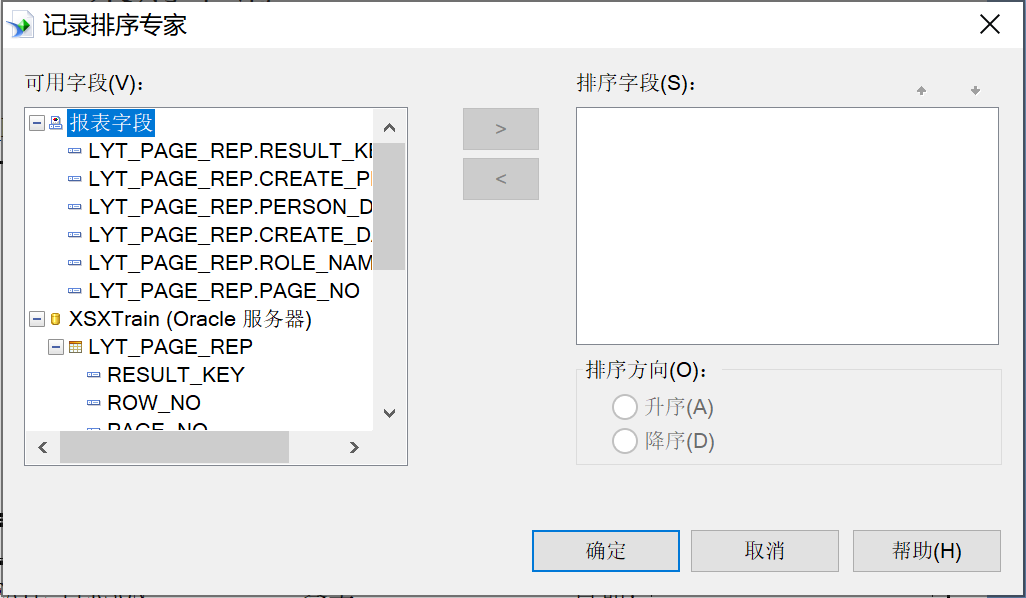
前台vscode/ webstorm打开QinzhouAPP

导入项目 npm install

ionic serve 启动

页面字段插入数据时按序插入

需要排序哪个字段就添加到哪边



钦州APP

待验收bug

app验收完成后 数量未写到erp上

验收单：7258 3 1 3

验收流程

待接收模块 erp内 采购申请 --->零件申请行--->验收等级来设置几级验收人

逻辑：固定资产，验收等级（）

非固定资产验收等级（）

采购--->接受--->采购订单接收

一个采购单会生成多个验收单（采购单上的物资不一定能全部送达验收，来一批验收一批，所以多单验收）

采购订单号、行号、下达号、接受号四个字段确定一个验收单

正常采购单产生接收单后，接收物品数量确定 后提交 则 采购订单接收页面数量随之改变

采购页面财务人员标识和经理标识对应右键签名

采购---需求---采购申请 申请时填写采购类型 ：固定资产/非固定资产

判断采购类型是否为固定资产也可在 采购---->订单---->采购单---->零件订单行----->会计组 查看类型

固定资产 到达验收已完成列表需要： 验收完成&签名完成

非固定资产 到达验收已完成列表需要： 验收完成

app验收结果处理：

验收已完成：包含验收完成的非固定资产 和 验收完成&签名完成的固定资产

验收未完成：包含 验收未完成的非固定资产 、验收未完成的固定资产、签名未完成的固定资产、验收&签名均未完成的固定资产

验收完成后状态会 变为 将检验:检验员检验，完成后状态变成将收到

将收到：此时可入库 即完成

采购申请--->采购订单--->采购接收单--->库存

申请单-->采购订单 （已下达）已到达

已接收

申请-->（报价）-->采购订单-->收据-->检验

采购申请---（请求报价）---采购订单----->下达供应商来采购订单--->送货（产生收货单）

物资管理操作手册：

物资编码申请；采购申请；物资采购；库房管理 ：物料入库、物料申请、退还已下发物料；库存查询操作 ：库存件信息查询、库存事务查询；物资盘点

库存-->编码-->物资编码申请

物资管理流程：物料申请 （生成） 采购申请 （审批） （下达 ）供应商 （根据采购申请单供货-->临时供货仓库） （生成）物料接收单 (一个采购单多个接收单) 验收人验收接收单 （入库）

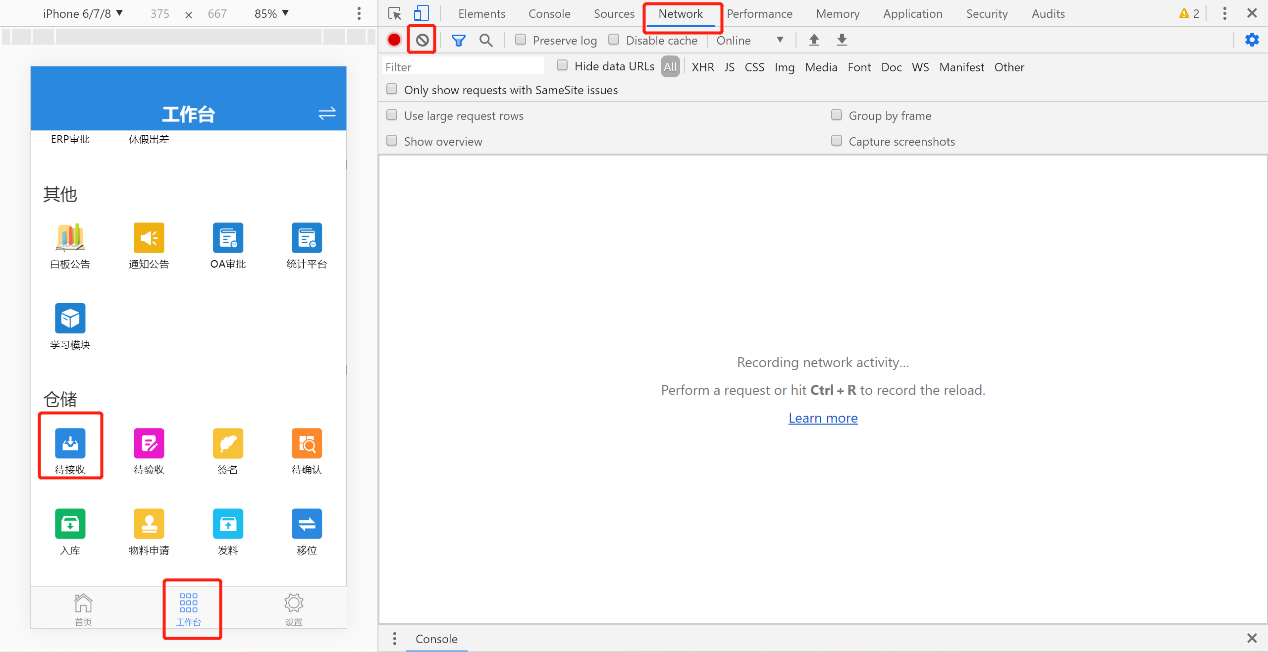
钦州移动仓储操作手册：

进入仓储管理模块----物料申请----待接收---待验收----待签名----待确认----待入库-----发料---移位----物料查询----退料----盘点

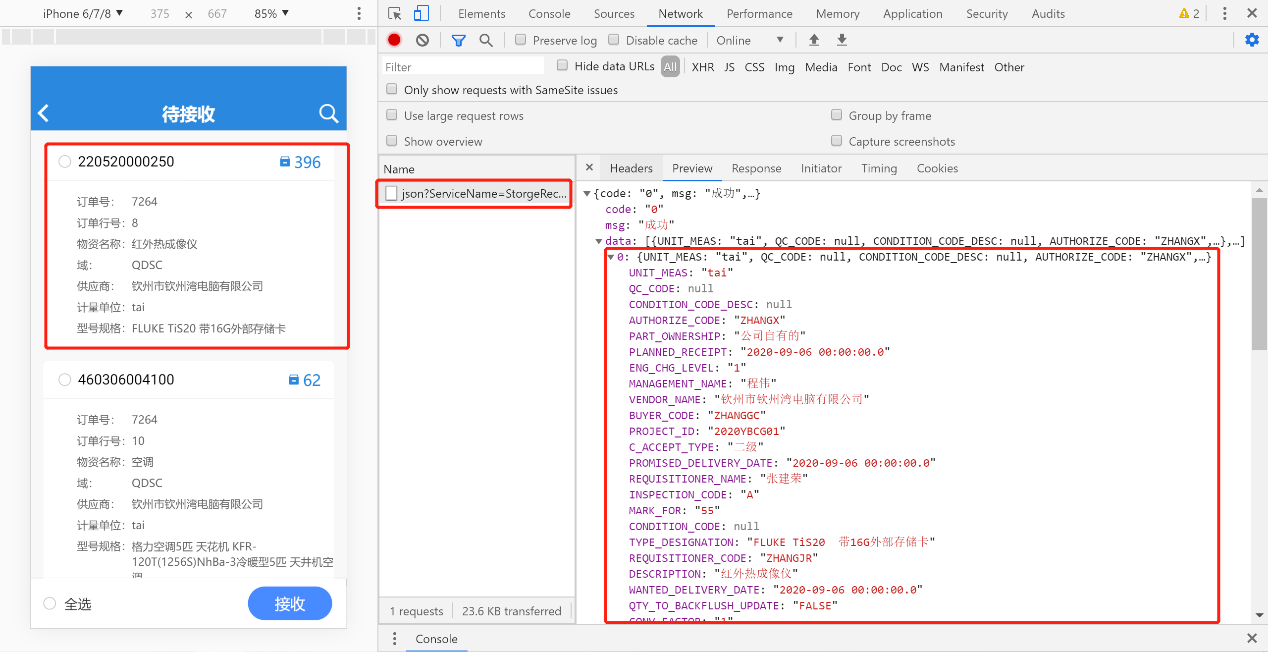
BUG查验流程：比如 待验收物资卡片 物资信息显示错误

启前台后台数据库

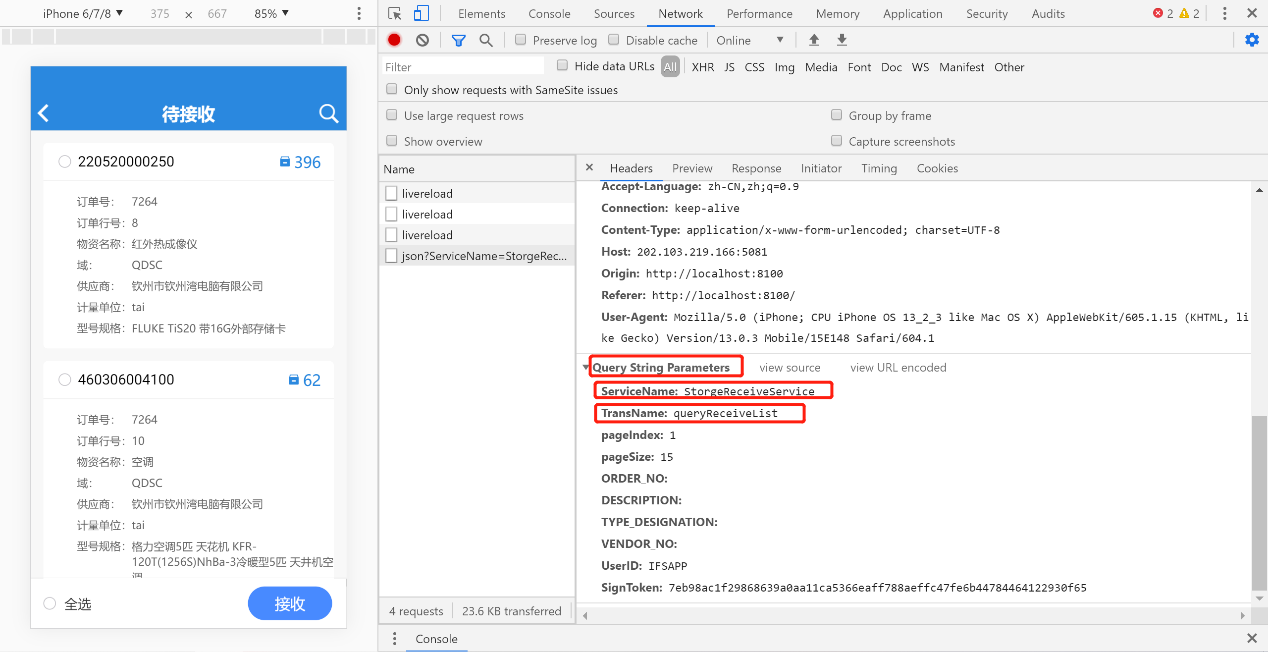
把network下的内容清一下，点进待接收模块



比如第一个资料卡信息（物资名称）出错，在name列找到json的详情，在preview项下，找到0号data行（资料卡编码从0开始）查看 物资名称的传入字段是 description ，查验它传输正确否，

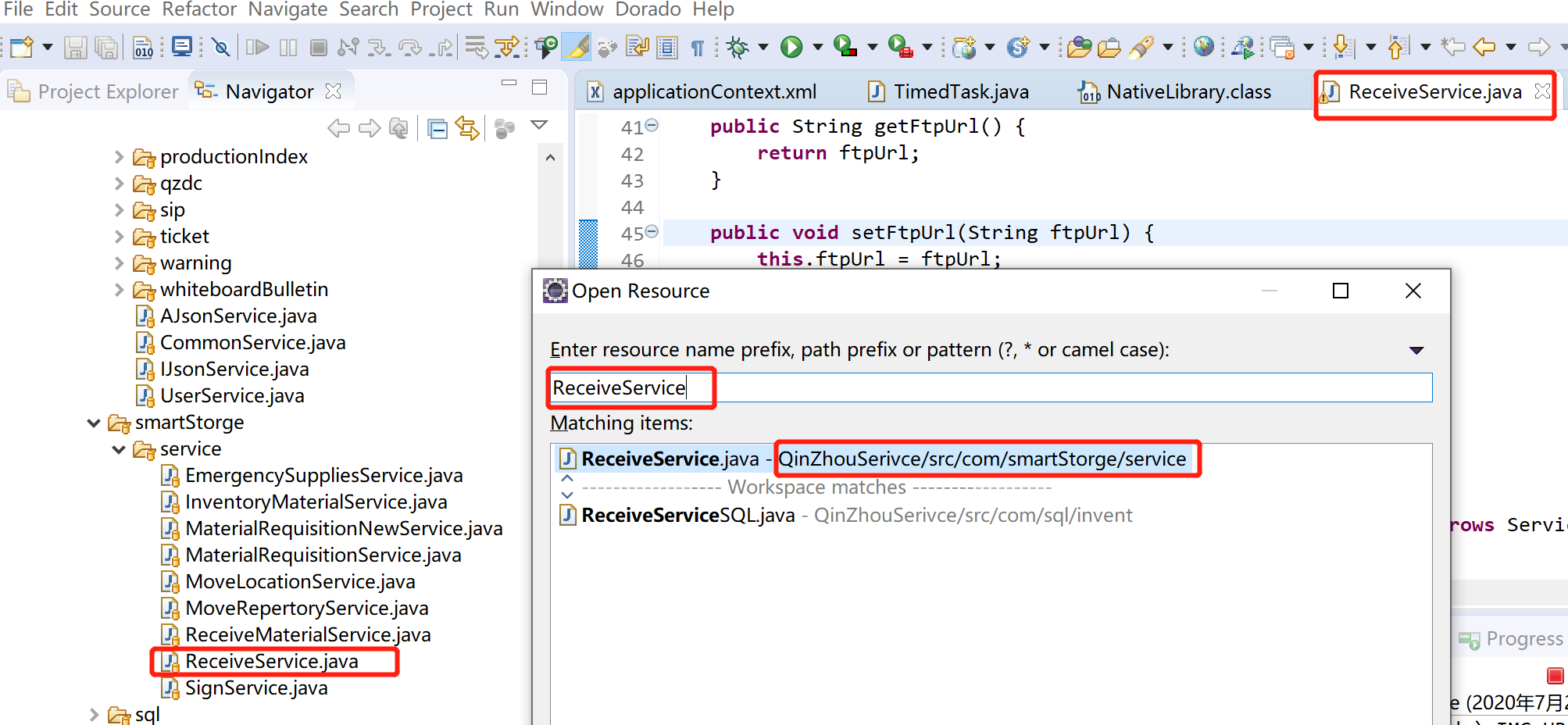


有问题切换页签到Headers，Query String Parameters 项 中的 ServiceName 是后端类名，TransName 是方法名



进后台，包显示和目录结构显示，window--show View --Navigator

Control +Shift+R 查找类名 ，打开类（查的时候把Store去掉）



在类中Control +F 查找方法，看看方法逻辑

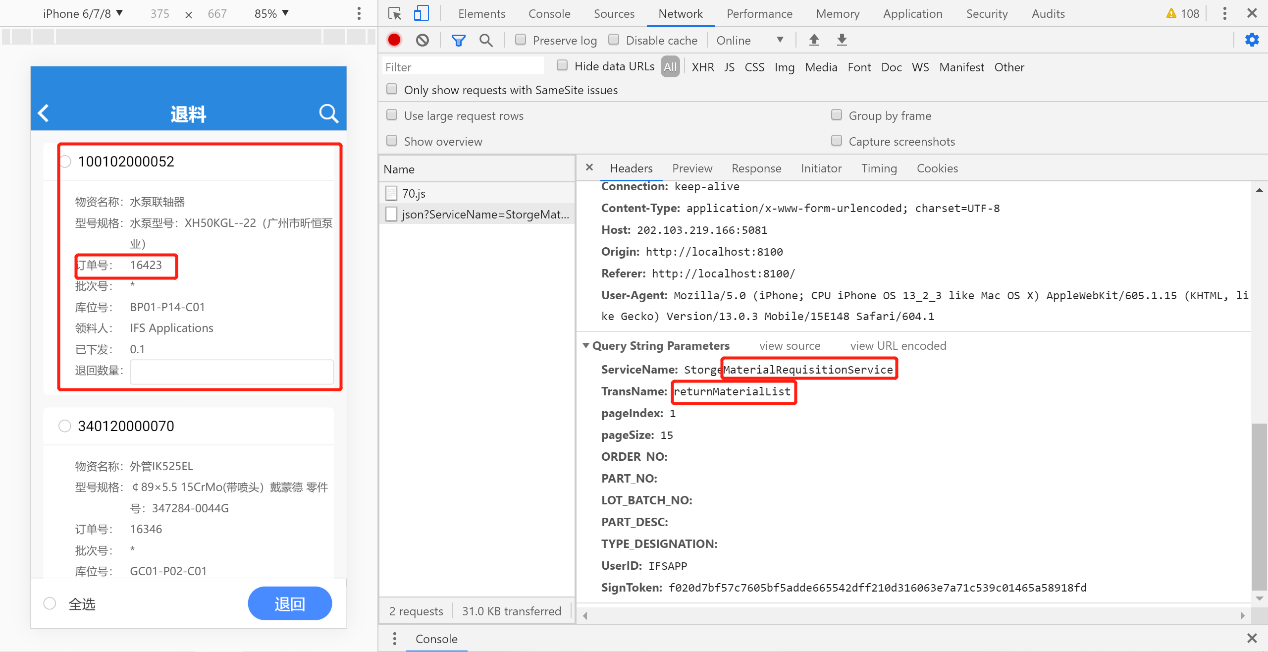
找到SQL语句拿到数据库中查验，看查到的数据对不对，排除SQL写错。



总有一步出错，解决就正常了

退料单按订单号排序

从app走到退料 ，找到 类和方法



点进去看sql

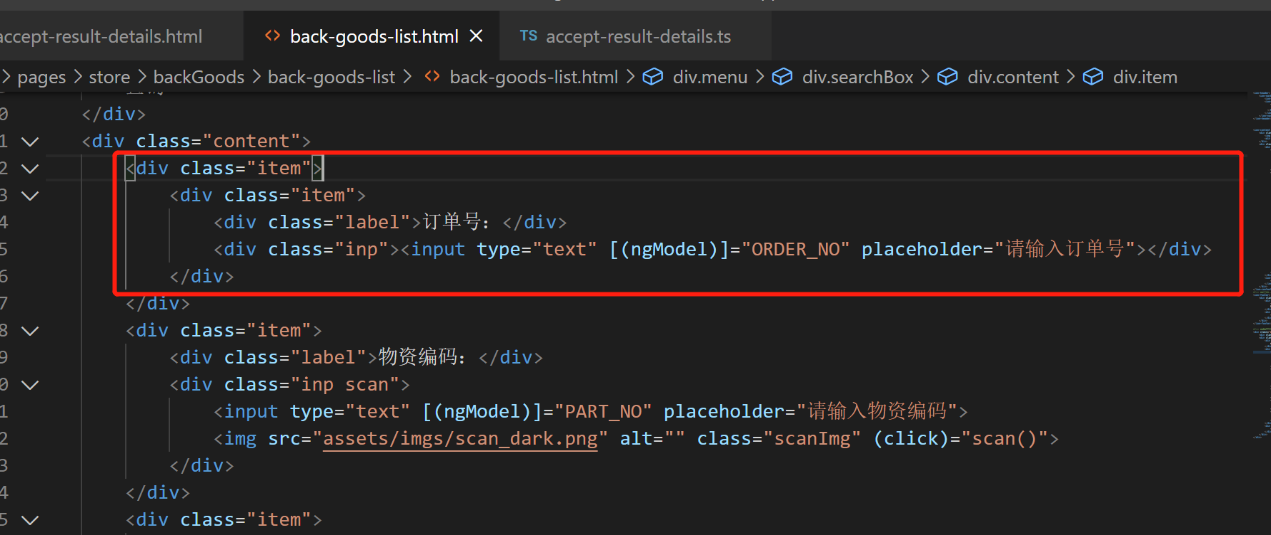


给查询SQL和分页SQL加上order by

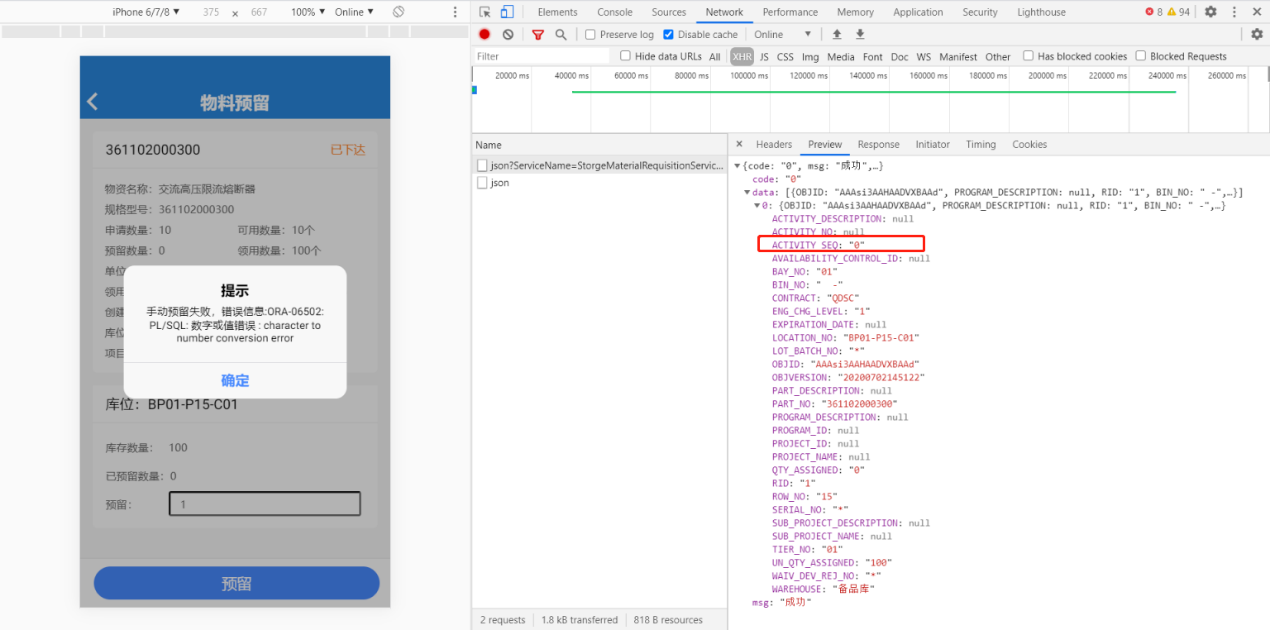


退料增加订单号搜索字段

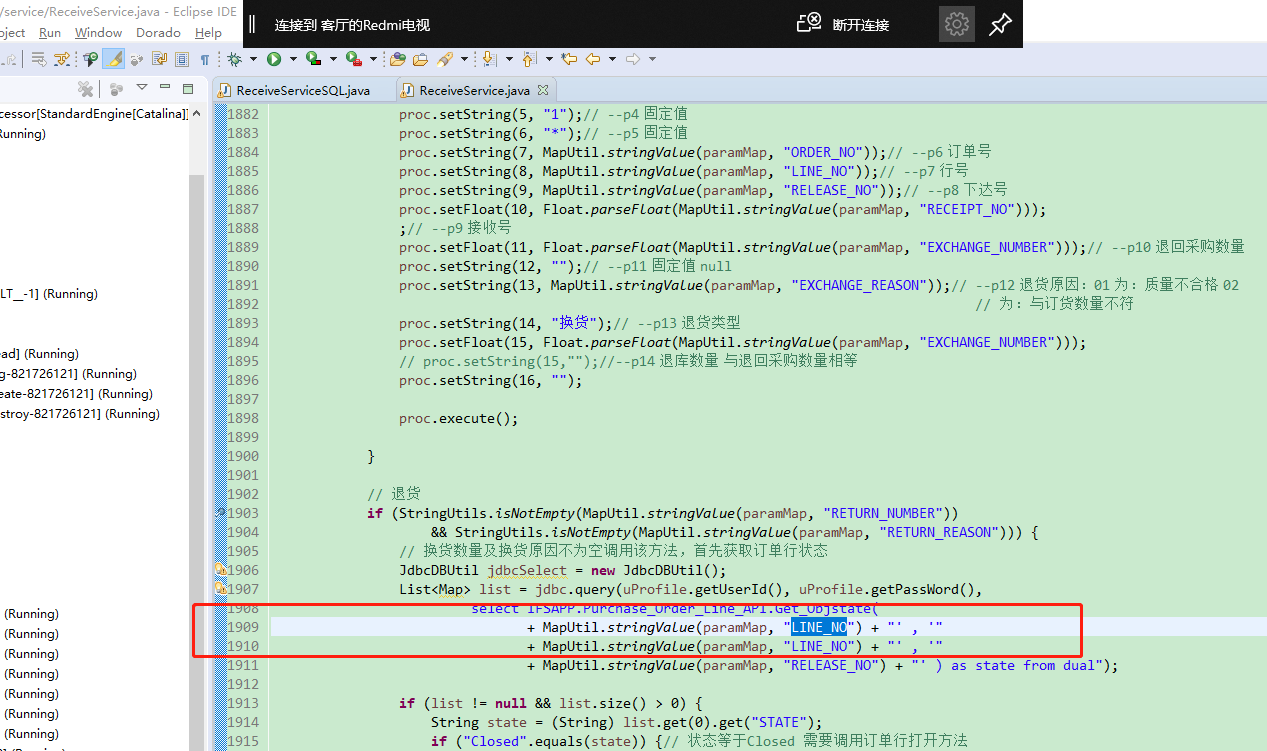
前台加div显示字段，传入order\_no 字段参数



物料预留 active\_seq字段前后台应该一致，不能为null

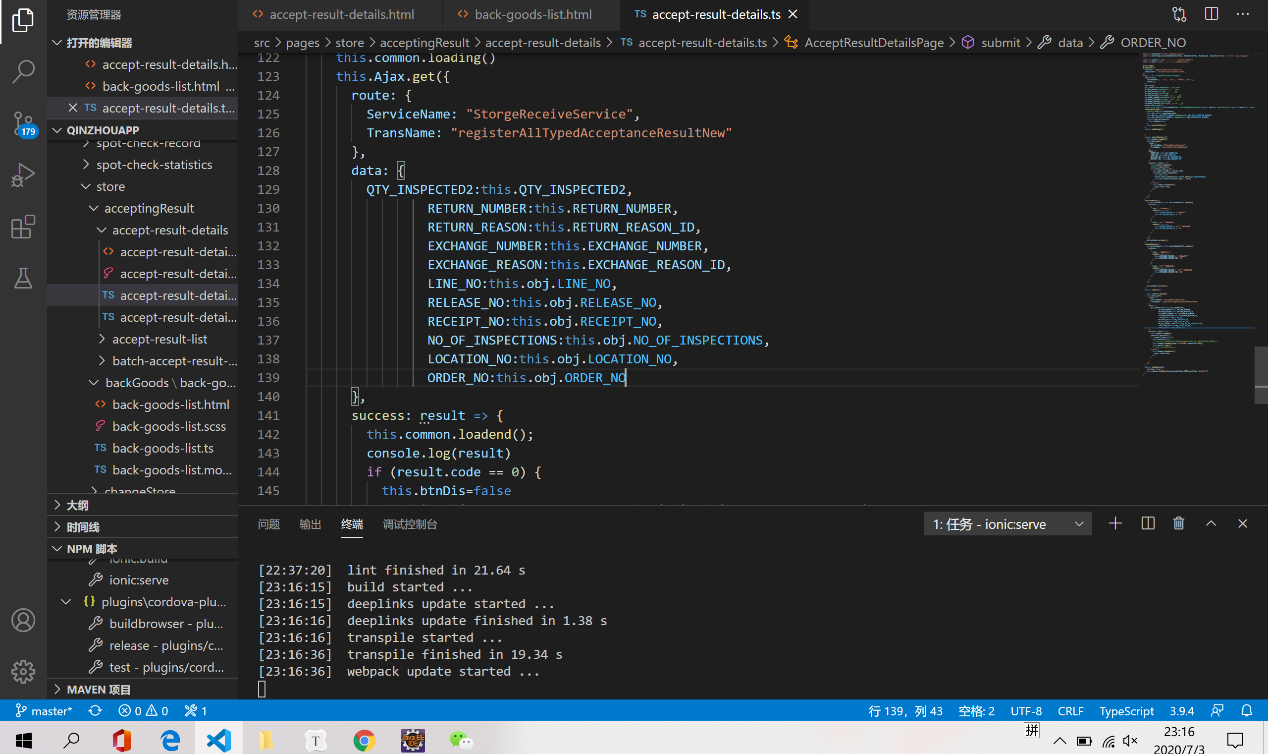


流程：先查类 查方法，然后console报错，找到自己的类，（debug启动）打断点，重新走流程，放到查到的字段上看值，然后查一样不。感觉是sql就把sql拿到数据库里查。要的结果和查的结果对比。改好把值传到前台，就好了

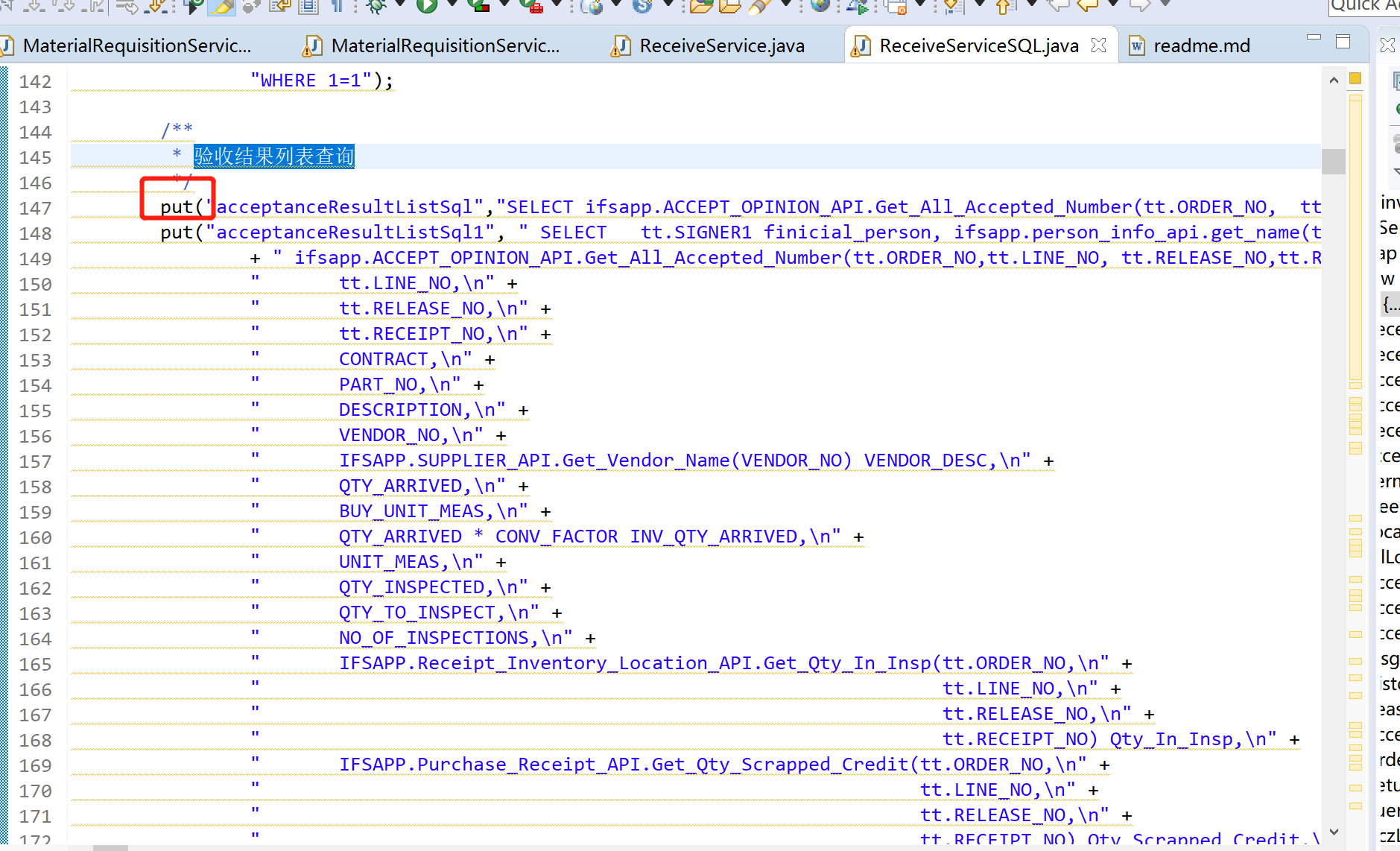


退换货数量没有写到erp，订单号传错了，order\_number,

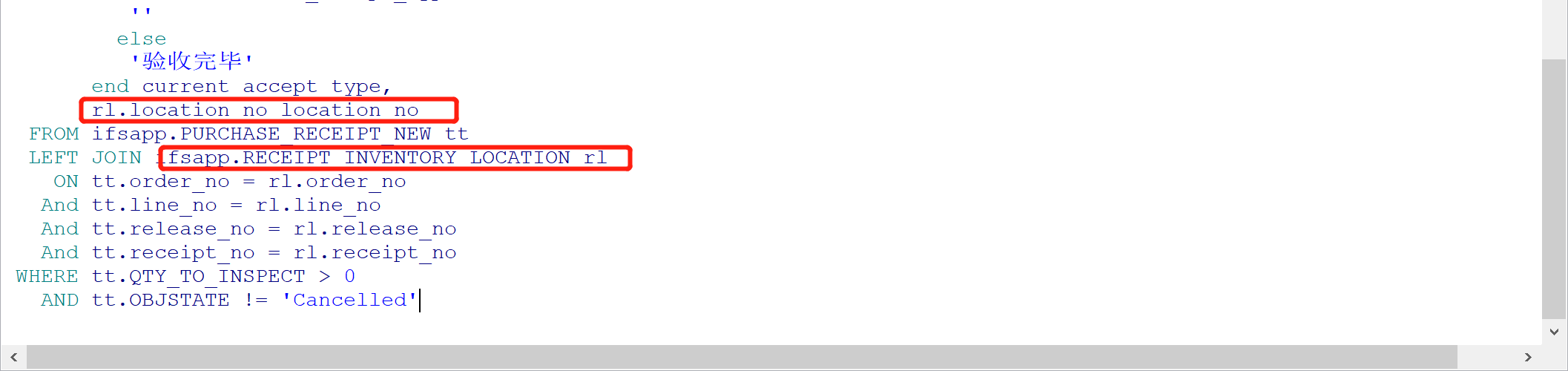
库位号location\_no 先在调用的存储过程中 查看是否有location\_no字段，没有加上，字段不能写死，后期难以维护，



前台和receiveService中拿到库位号的字段匹配一致



在ReceiveServiceSQL中 加join 联查



业务流程复盘



物料申请-->发料-->待验收-->待确认(验收未完成)-->签名（固定资产)-->待确认（验收已完成，确认）-->入库-->退料、查询、移位、盘点