目录

[一. 要学会的技术点 2](#_Toc29377)

[二. JQuery 2](#_Toc17318)

[1. 项目运行 2](#_Toc28252)

[1. 打开本地项目配host： 2](#_Toc18487)

[2. 知识点 3](#_Toc22052)

[1. jq Ajax 请求 参数形式： 3](#_Toc7572)

[2. 下拉表单 数据叠加问题 ：加这个,removeChild()不太行。 3](#_Toc26763)

[3. Input的Password密码在消息头header中能看到！：（消息头中密码要加密显示 md5） 4](#_Toc1281)

[4. session超时处理 跳转登录页面 （+ 页面关闭后，无论访问哪里，先跳登录） 4](#_Toc21499)

[session储存： 7](#_Toc25772)

[5. 阻止点击事件冒泡 和 阻止默认点击事件 7](#_Toc14763)

[6. Jq添加属性 8](#_Toc26024)

[7. jq jstree 动态选中控制。 9](#_Toc24630)

[三. Vue-element-admin项目 9](#_Toc13996)

[笔记布局： 9](#_Toc23509)

[笔记步骤：（程度：可以拿笔记去当讲师） 9](#_Toc16876)

[经验： 10](#_Toc24931)

[规范 10](#_Toc22142)

[目的 10](#_Toc25241)

[1 开始工作环境搭建 10](#_Toc2606)

[1. 高通入职 10](#_Toc17708)

[1. 软件安装 2019/3/18 10](#_Toc32432)

[2. 软件安装 2019/3/20 11](#_Toc21409)

[3. 2019/4/4 11](#_Toc19846)

[1. 在webstorm上debug 11](#_Toc15904)

[4. 2019/4/15 12](#_Toc14372)

[1 与后台联调，对数据 12](#_Toc29328)

[2. 开发高度提醒：关于接口文档 12](#_Toc26128)

[2 Vue-element-admin项目 启动 12](#_Toc3766)

[1. 跳过登录（小心缓存） 12](#_Toc2313)

[2 去掉左右v-handle：（权限指令 控制是否隐藏） 13](#_Toc1072)

[3. 登录联调 连后端ip完成登录 13](#_Toc16262)

[4. 跳过路由权限 （跳过路由权限过滤） 15](#_Toc1422)

[2 项目知识点 16](#_Toc4105)

[1. 项目流程说明 16](#_Toc19661)

[2. 项目注意的点 16](#_Toc27668)

[1. 文档要清楚 16](#_Toc3835)

[2. 评估时间要充裕 16](#_Toc24516)

[3 问题：？？？ 16](#_Toc19932)

[1. 单点登录 是如何做到的？？？ 16](#_Toc31591)

[4. 技术点 17](#_Toc32703)

[1. Vue Input 输入控制：input输入数字大于1 保留小数点后2位 17](#_Toc11902)

[2. 发布加密包（看不到浏览器webpack代码） 把 productionSourceMap: true 改成false就可以？ 17](#_Toc1761)

[3 权限 17](#_Toc18440)

[4 vue 下载模板--上传表格--生成列表 （解析excel数据） 看项目代码 18](#_Toc32021)

[6. 封装公共组件：分线程获取数据/20个线程 19](#_Toc11796)

[7. 封装公共函数 19](#_Toc15759)

[8.表格-加行-重置-表格数据提交 21](#_Toc12008)

[8. 动态组件显示 21](#_Toc9717)

[9. 存md5等api进vue.propoty 22](#_Toc8245)

[10 自定义指令 22](#_Toc4400)

[例子1——按钮点击波纹扩散：v-waves 22](#_Toc25916)

[例子2——权限控制：v-handle=”” 22](#_Toc24979)

[11 字符串部分隐藏显示 22](#_Toc31438)

[5. 疑难bug （耗费大量时间解决的坑） 23](#_Toc23879)

[1. 循环机制 （涉及引用变量的循环，注意：放在外边还是里面！） 23](#_Toc24714)

# 要学会的技术点

|  |  |
| --- | --- |
| Websocket |  |
| Rem布局 |  |
| Js封装 交互组件 |  |

# JQuery

## 项目运行

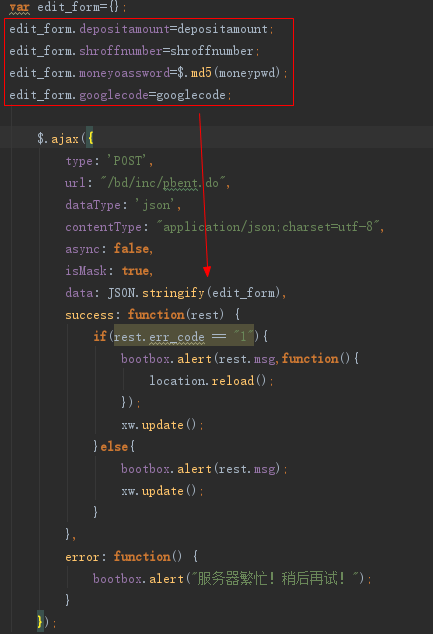
### 打开本地项目配host：

然后关闭nginx和node后台服务，然后输入mer.gt.com就可以



## 知识点

### jq Ajax 请求 参数形式：



### 下拉表单 数据叠加问题 ：加这个,removeChild()不太行。



### Input的Password密码在消息头header中能看到！：（消息头中密码要加密显示 md5）

①后端一般默认接受md5加密类型密码（与后端配合好）

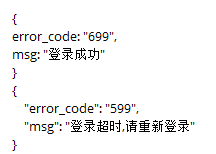
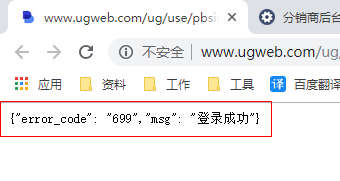
②前端这样设置参数

### session超时处理 跳转登录页面 （+ 页面关闭后，无论访问哪里，先跳登录）

需求描述： 1.超时后：输入[ugweb.com/index.html](http://www.ugweb.com/login.html)也要调到登录

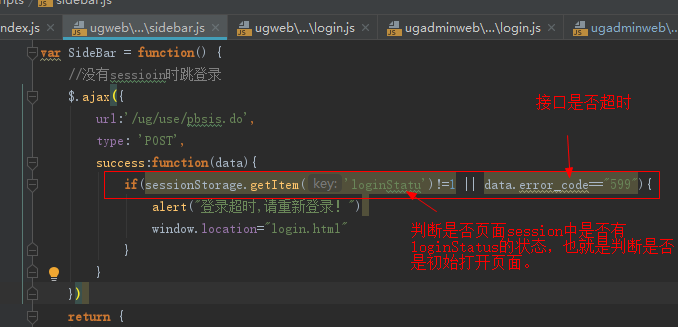
1. 点击页面：跳转登录，登录后再回来原页面
2. 登录后，新窗口打开www.ugweb.com，直接显示页面（而不应显示登录页www.ugweb.com/login.html）

①做接口：（有漏洞）

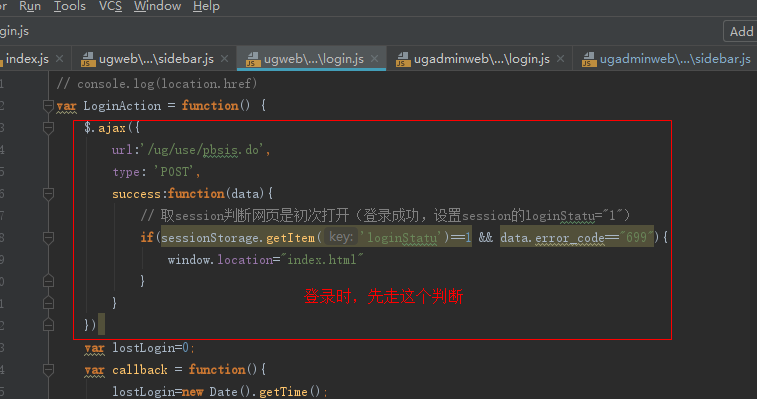
 

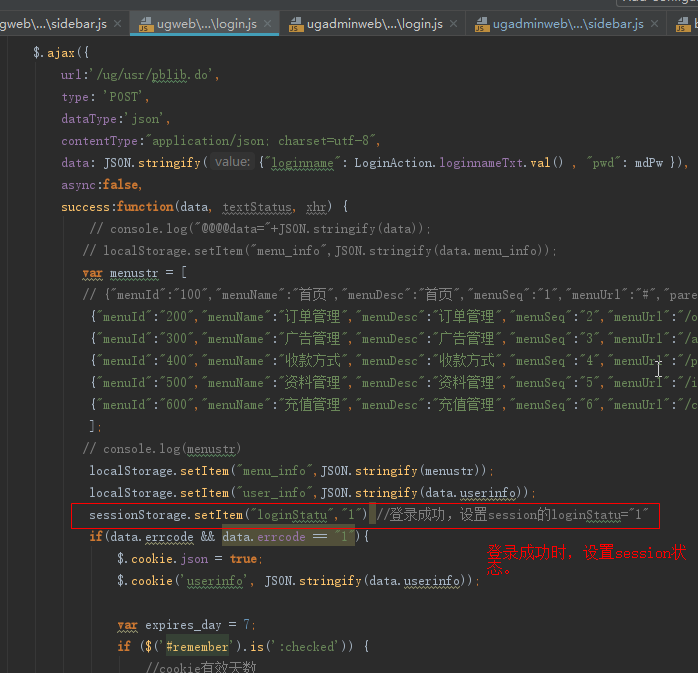
1. 公共js变动时，请求接口：

1.访问知道超时—>跳登录页2.是否是初始打开网页->是->跳登录



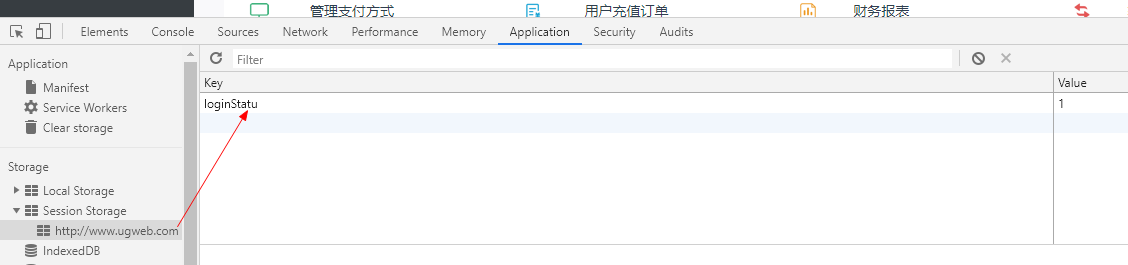
1. Login页请求接口：访问知道是否登录—>跳登录页





1. 关闭页面后，访问login不能跳到index去，应重新登录。

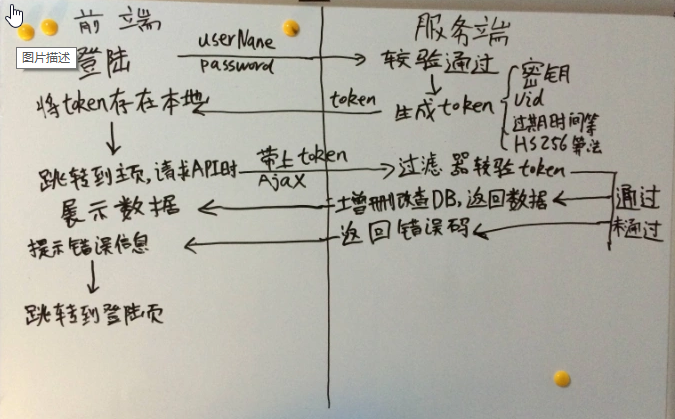
（利用关闭页面session清除，所以登录成功时在session里做个状态：所有页面的公共js读取不到session这个状态，就跳到登录页）



1. 防止页面后退

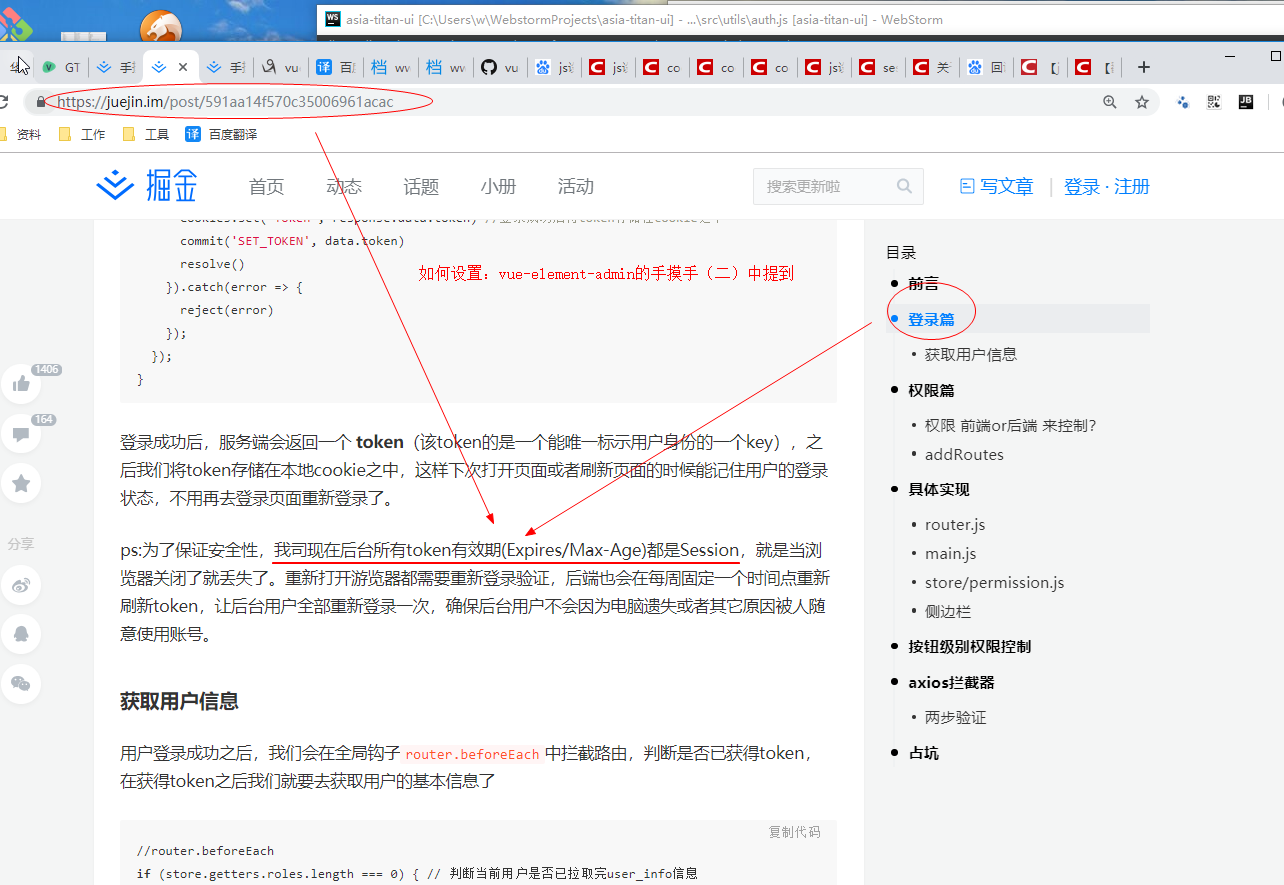
<script language="javascript">  
 //防止页面后退  
 history.pushState(**null**, **null**, document.URL);  
 window.addEventListener('popstate', **function** () {  
 history.pushState(**null**, **null**, document.URL);  
 });  
</script>

②token方法：



③？？？我司现在后台所有token有效期(Expires/Max-Age)都是Session，就是当浏览器关闭了就丢失了。

参考：<https://juejin.im/post/591aa14f570c35006961acac>



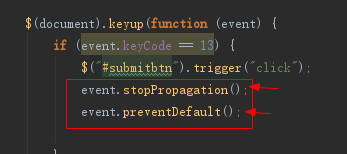
参考：js设置cookie过期时间为session

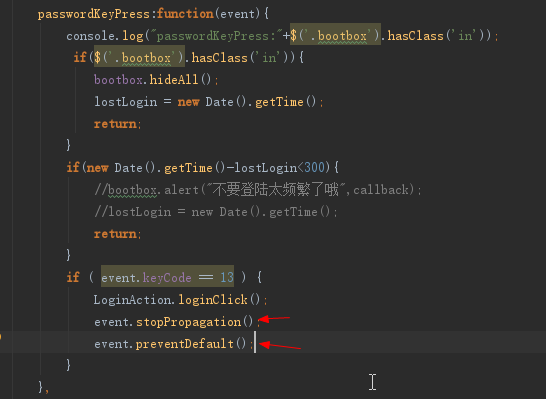
<https://blog.csdn.net/haoxinqing9698/article/details/7550804>

### session储存：

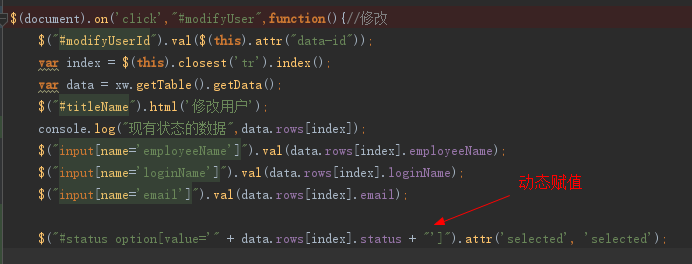
sessionStorage.setItem('getMenu', obj)//存  
sessionStorage.getItem('getMenu')//取  
sessionStorage.removeItem('getMenu')//删

### 阻止点击事件冒泡 和 阻止默认点击事件





### Jq添加属性

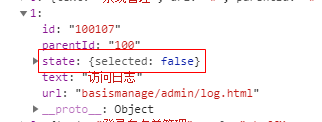




### jq jstree 动态选中控制。







# Vue-element-admin项目

## 笔记布局：

1 标题

2 知识点

3 代码截图

4 提示 案例 等~

## 笔记步骤：（程度：可以拿笔记去当讲师）

1. 晚上：git整理。
2. 早上：。
3. 关键词

熟悉需求：需求是什么！ （你会什么？ 我会制表格）

## 

## 经验：

1. 出错一般上是代码写错

## 规范

|  |  |
| --- | --- |
| **类型** | **关键词** |
| 未解决 | ？？？ |
|  | 浅灰背景：搜索关键词 |
| 待办 | \*\*\* |

## 目的

**熟悉产品整套开发流程及开发规划！**

# 1 开始工作环境搭建

## 高通入职

### 软件安装 2019/3/18

1. node 安装：去官网下载包安装，8.0以上
2. npm 安装：（node自带了）但配置环境变量：

https://www.cnblogs.com/goldlong/p/8027997.html

1. 谷歌浏览器下载
2. 纸飞机
3. Word下载安装
4. 搜狗输入法——设置截图快捷键ctr+shift+c
5. 编辑器：webstorm VScode
6. 设置编辑器快捷键（笔记2）
7. 安装git：官网下载——webstorm-vcs/Git/clone ——克隆代码

(△拉不到代码：自个工网ip（上网查）找运维绑代码库）

1. 安装解压软件：2345解压
2. 浏览器身份验证器：问别人要秘钥如：XUDCPVDP3AA7DTGG,然后在验证器里添加

### 软件安装 2019/3/20

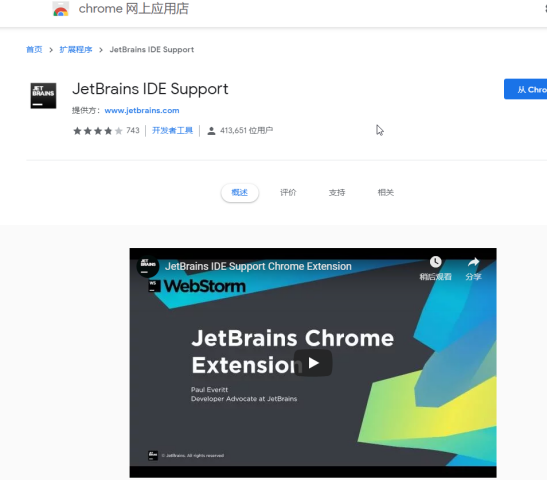
1. SwitchHosts 安装
2. Nginx 下载安装 问别人要配置文件nginx.conf——直到搞通代码调试 （**清缓存...**）

### 3. 2019/4/4

#### 在webstorm上debug

安装谷歌插件：

<https://chrome.google.com/webstore/detail/jetbrains-ide-support/hmhgeddbohgjknpmjagkdomcpobmllji?utm_source=chrome-ntp-icon>



参考文档：https://blog.csdn.net/u012212157/article/details/78095917

### 2019/4/15

#### 1 与后台联调，对数据

#### **开发高度提醒**：关于接口文档

开发文档问题：

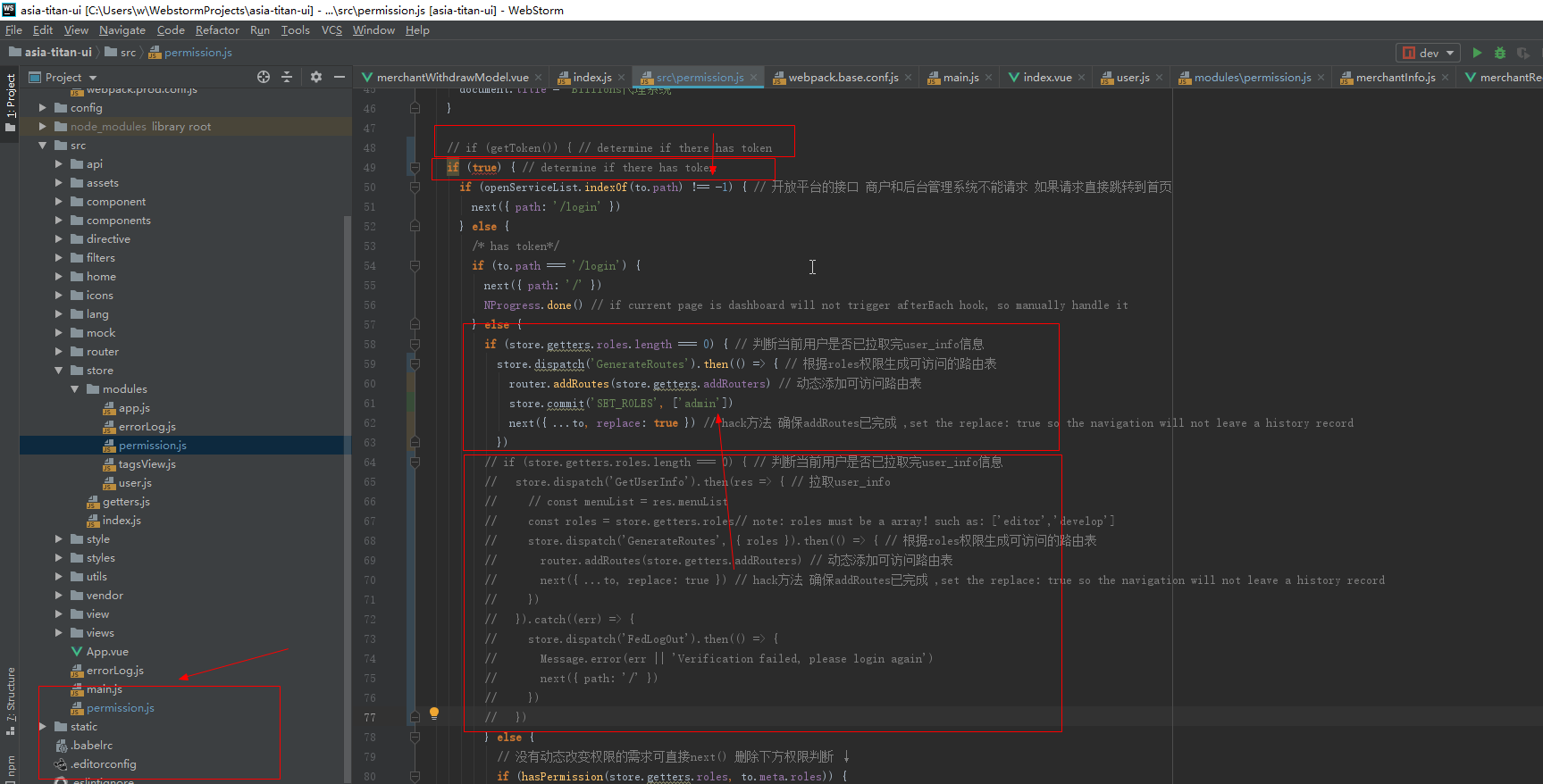
1 类型分不清的：md5 string number 传错参数后台有时返回服务器异常返回时根本不知道是那个参数类型错

2 不要直接返回参数，什么注释都没有：好比amount，数量。你不说是金额我哪知道后端在搞什么鬼业务。这步以后不能省，前端没法活，理解总有出入，猜来猜去的，根本干不了！

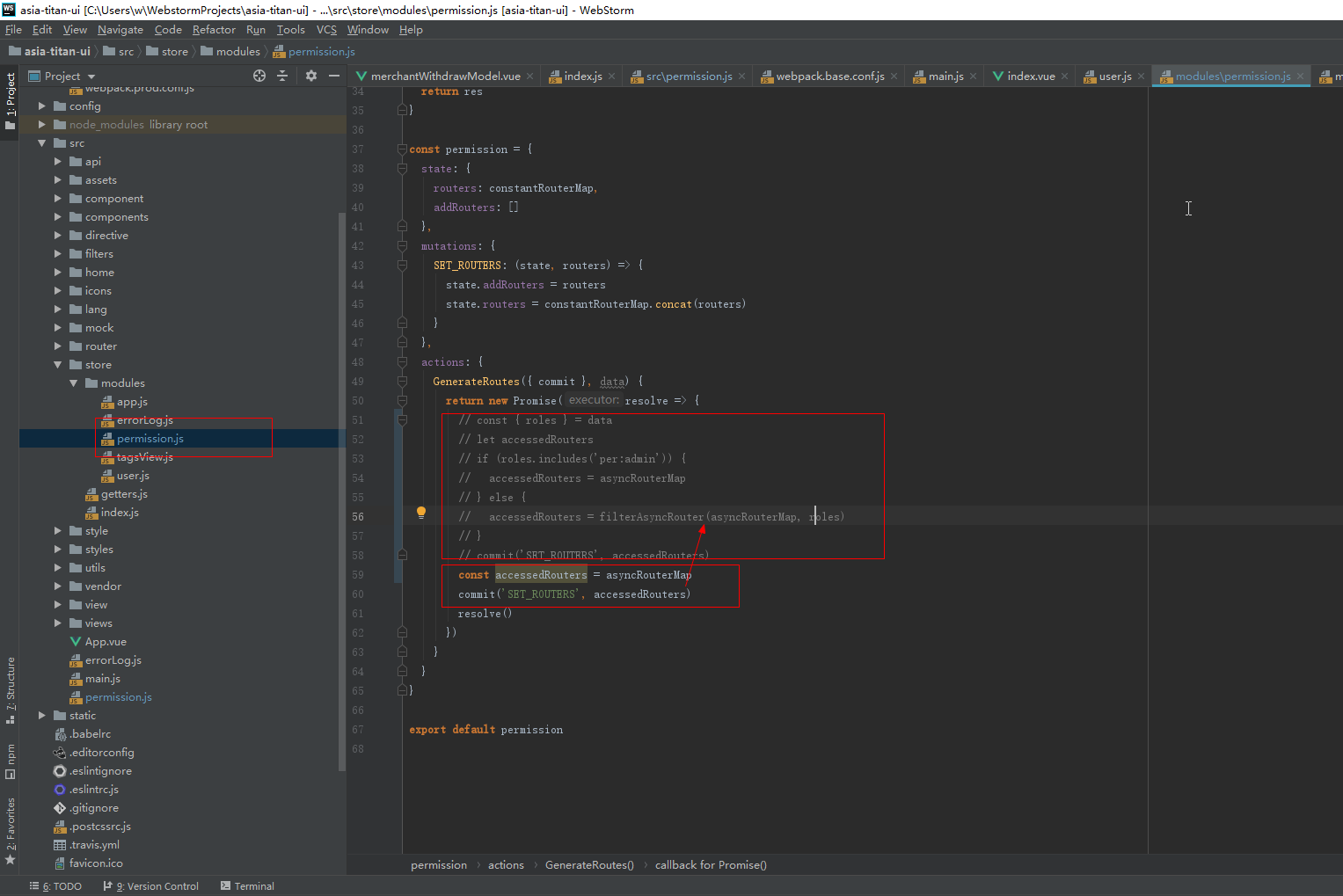
## 2 Vue-element-admin项目 启动

### 跳过登录（小心缓存）

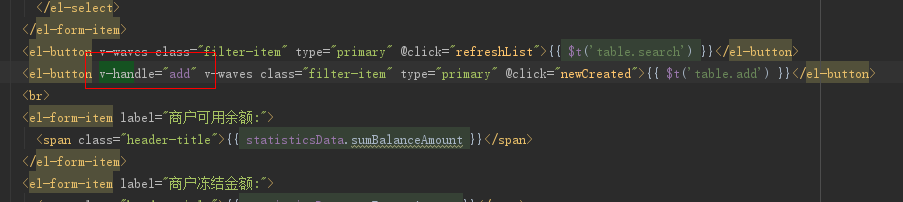
1.



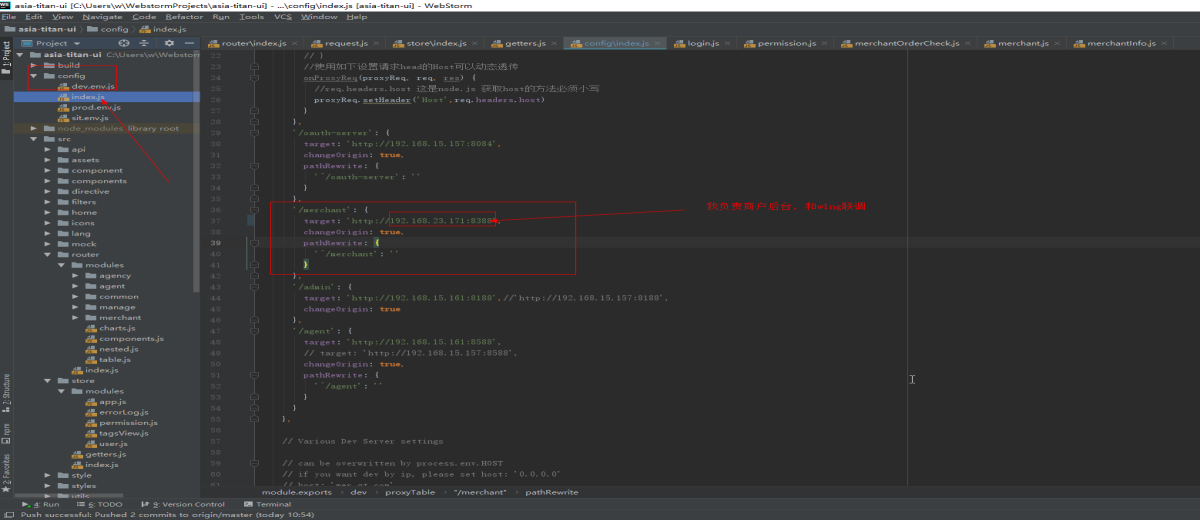
2.



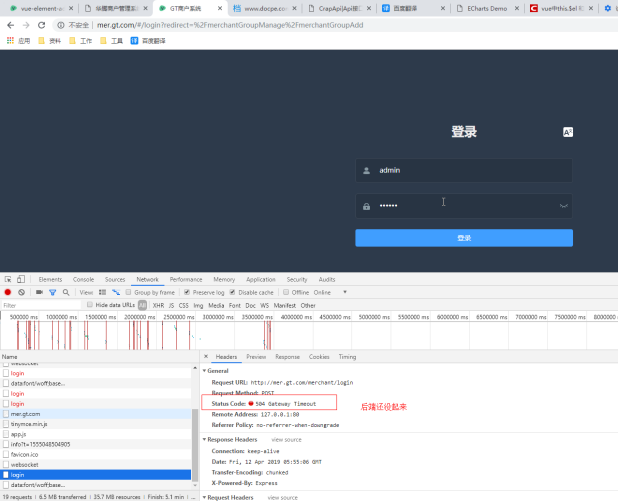
### 2 去掉左右v-handle：（权限指令 控制是否隐藏）



### 3. 登录联调 连后端ip完成登录

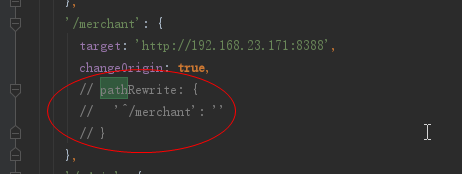


后端还没起来：



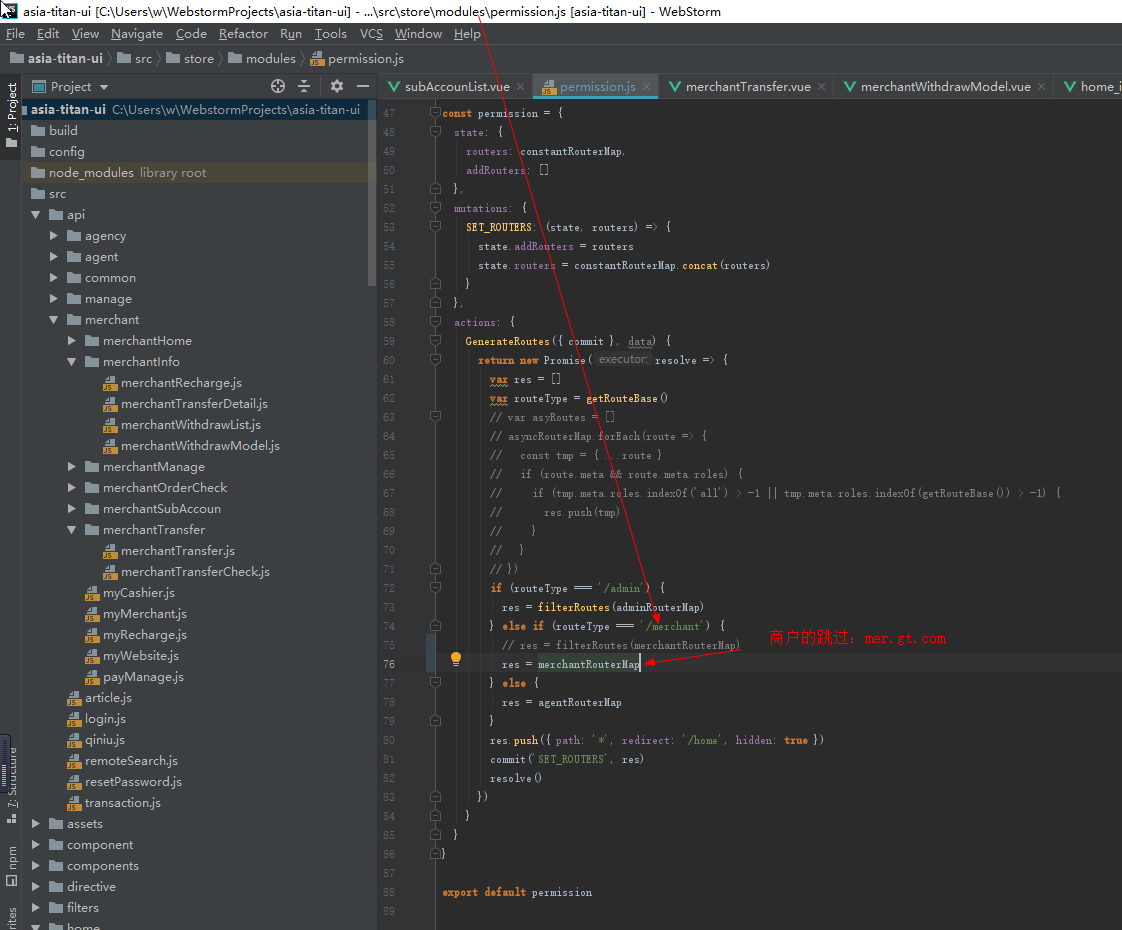
原因是重写了接口！：如下替换或直接注释。。

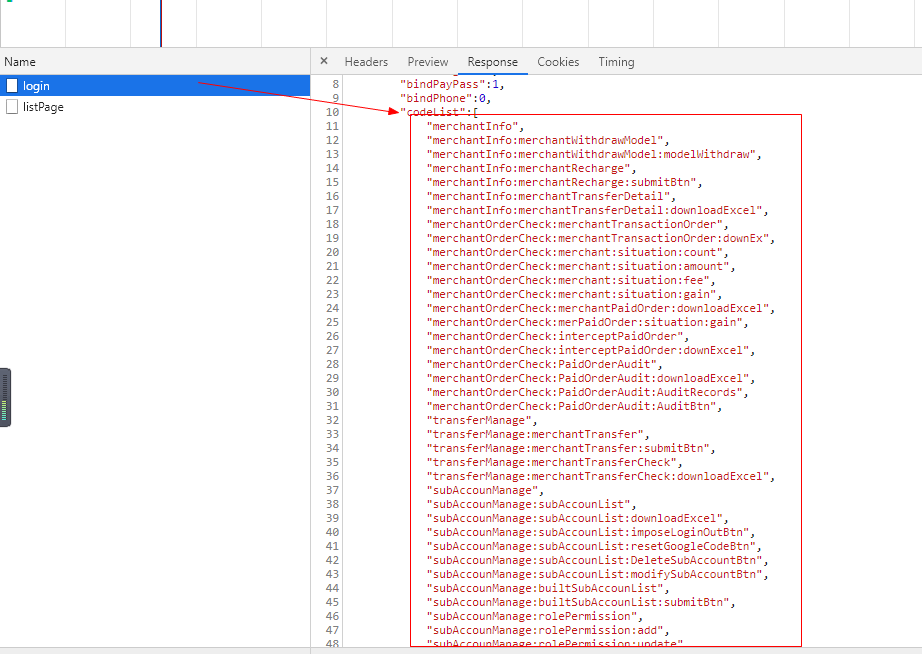
直接注释





### 4. 跳过路由权限 （跳过路由权限过滤）





# 2 项目知识点

## 项目流程说明

项目0到1

项目：后台重构

1. 功能模块——/高通后台重构/功能模块划分图
2. 设计图——/高通后台重构/代理后台.png
3. 进程计划——/高通后台重构/新项目重构计划.xlsx
4. 框架搭建——
5. 页面编写——

## 项目注意的点

### 文档要清楚

### 评估时间要充裕

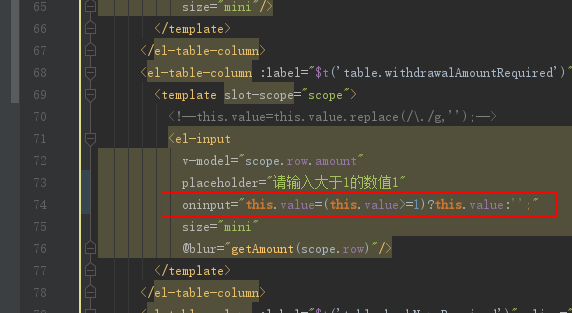
避免马虎代码，前后端推卸责任产生矛盾

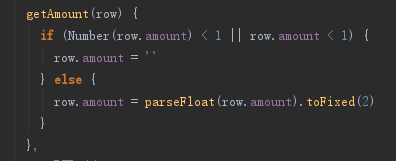
## 3 问题：？？？

#### 单点登录 是如何做到的？？？

## 4. 技术点

### 1. Vue Input 输入控制：input输入数字大于1 保留小数点后2位





### 发布加密包（看不到浏览器webpack代码） 把 productionSourceMap: true 改成false就可以？

路径：config/index.js

### 3 权限

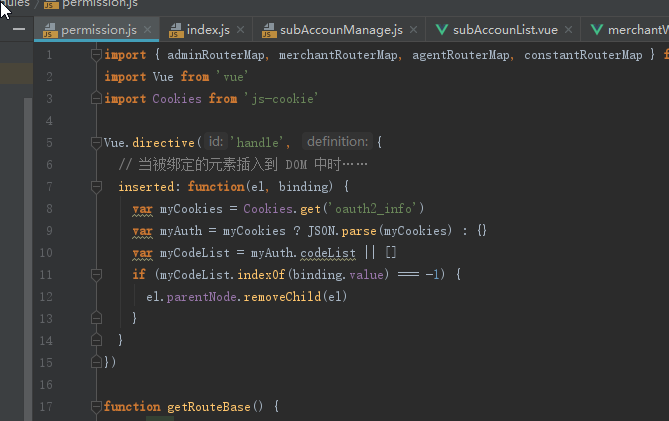
1 路由权限

**if** (routeType === '/admin') {  
 res = filterRoutes(adminRouterMap)  
} **else if** (routeType === '/merchant') {  
 res = merchantRouterMap  
} **else** {  
 res = agentRouterMap  
}

2 按钮级权限 做指令控制

例子：

v-handle="add"



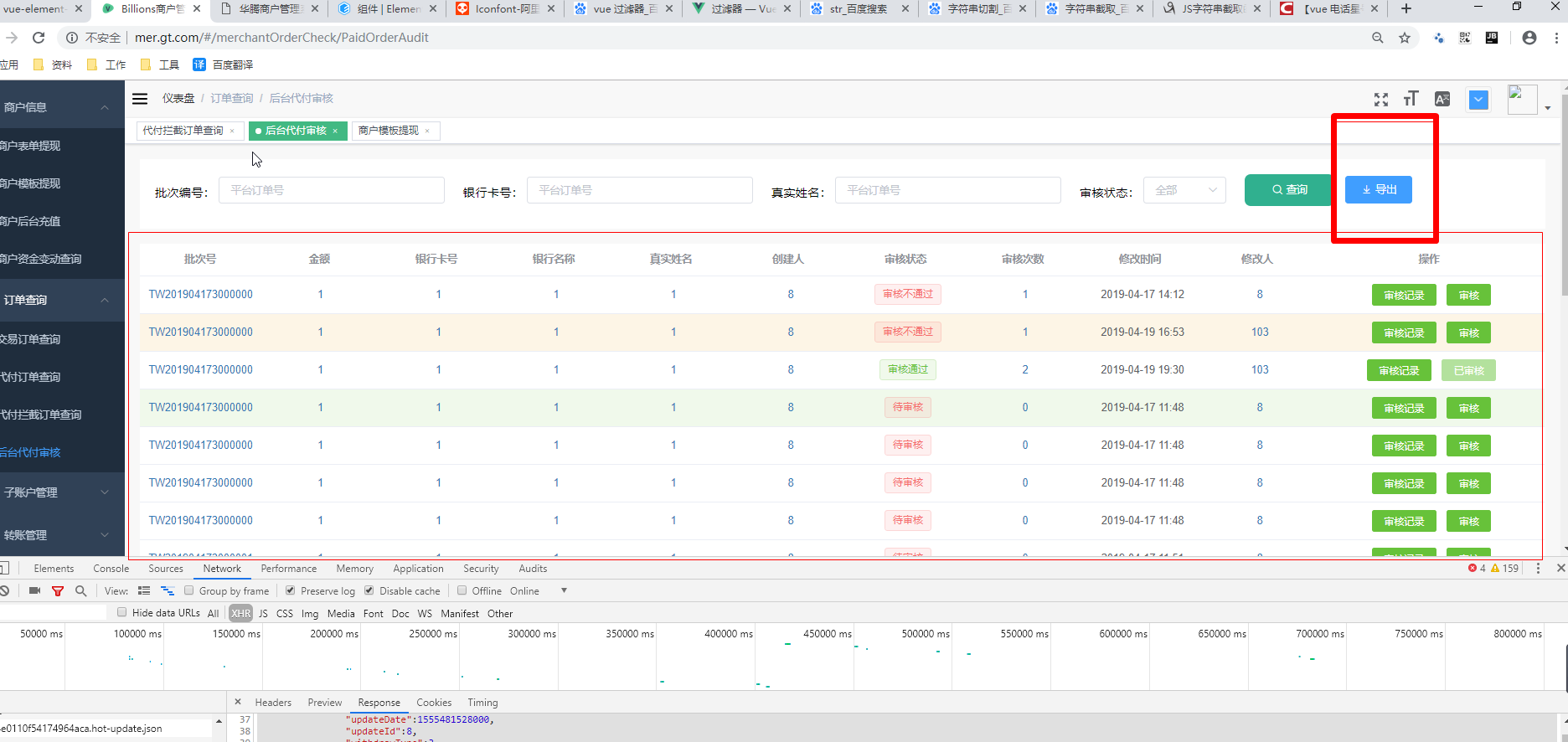
### 4 vue 下载模板--上传表格--生成列表 （解析excel数据） 看项目代码

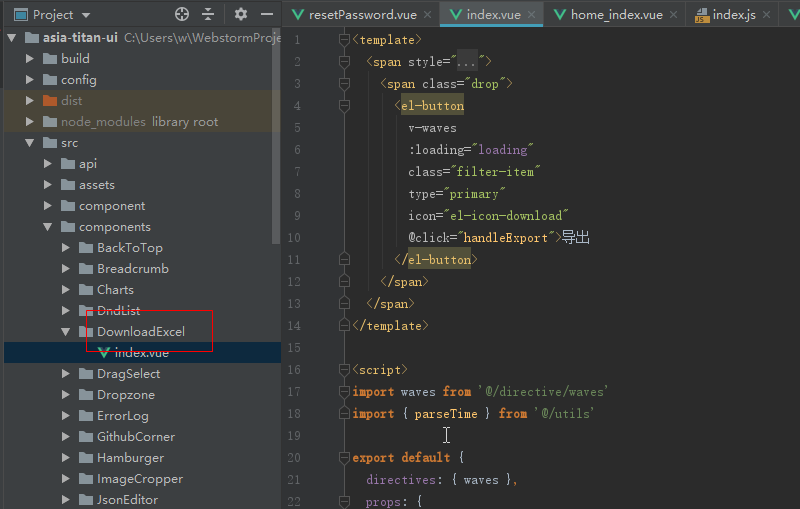
excel解析导入功能

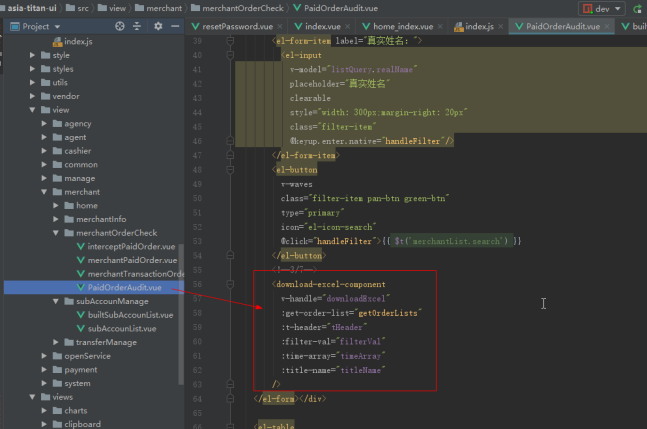


### 封装公共组件：

#### 经历：①分线程获取数据/20个线程







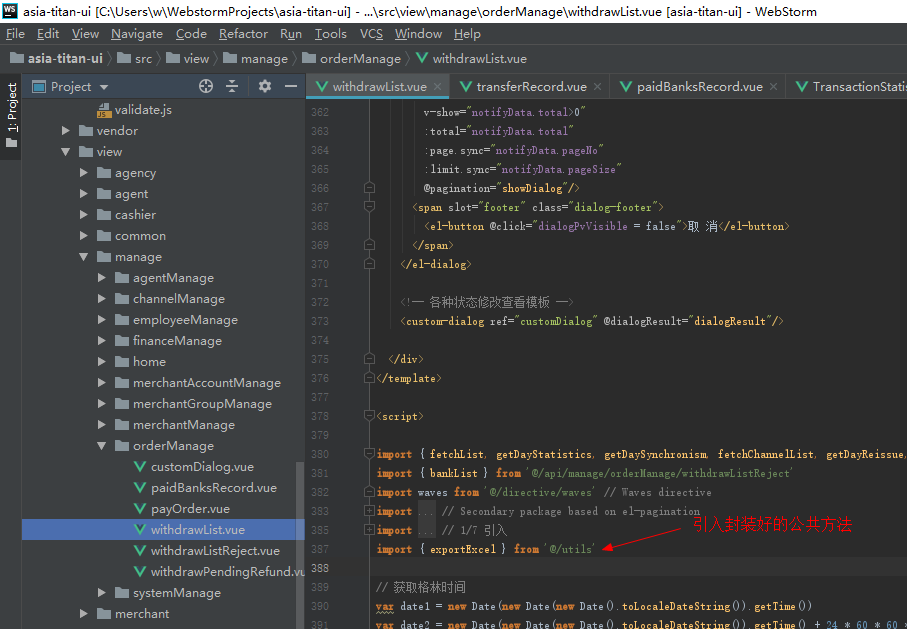
#### ②换成每个请求小于300条，每次最多发3个请求

思路：公共组件条用父组件方法——父组件返回公共函数结果（promise）..

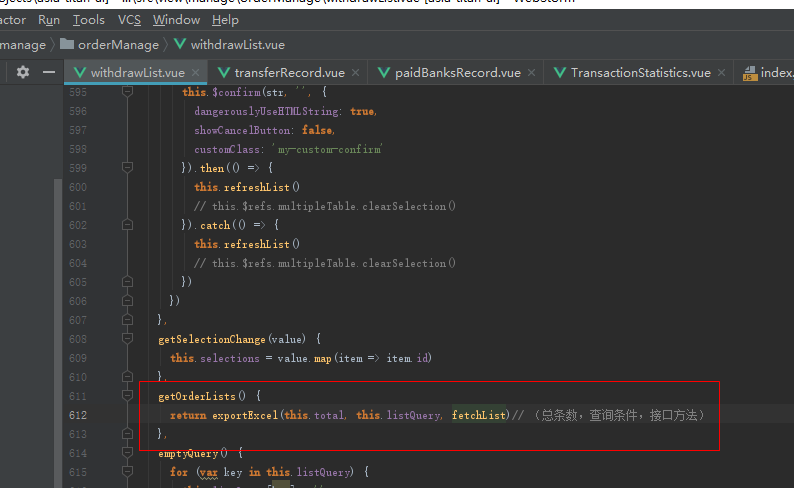
src/view/manage/orderManage/withdrawList.vue

src/utils/index.js

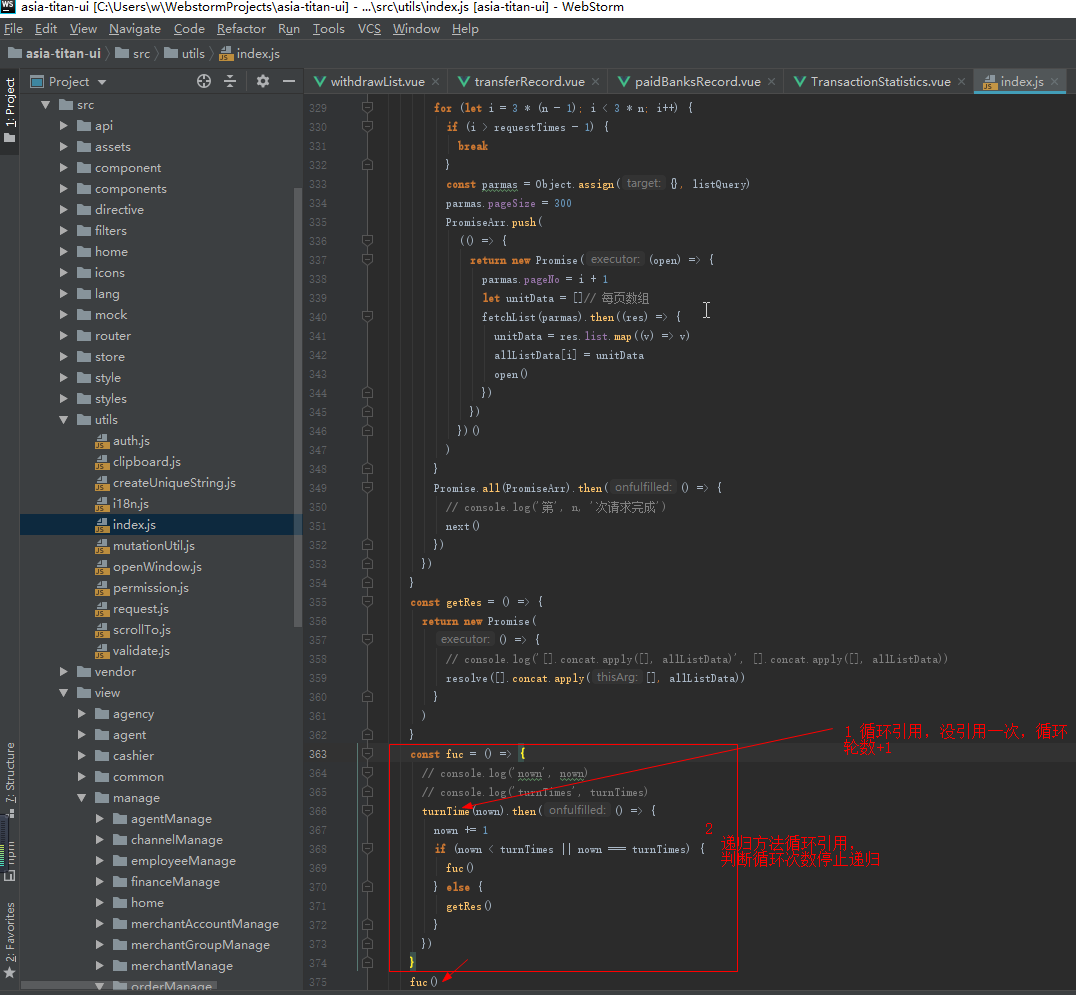
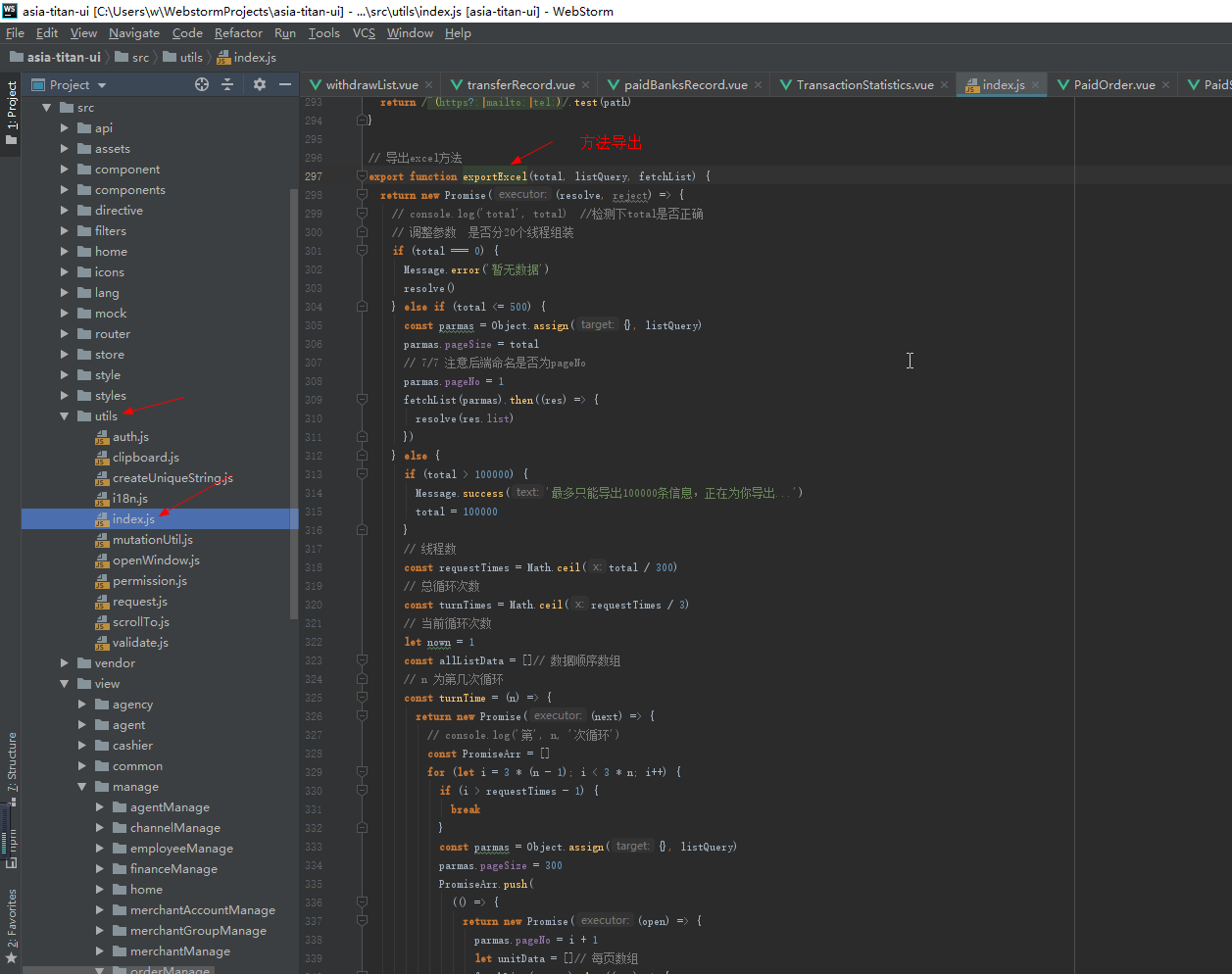
引用



调用

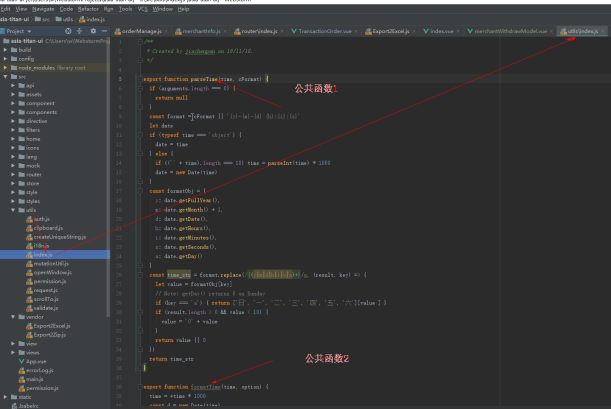


封装、导出（异步嵌套、递归）

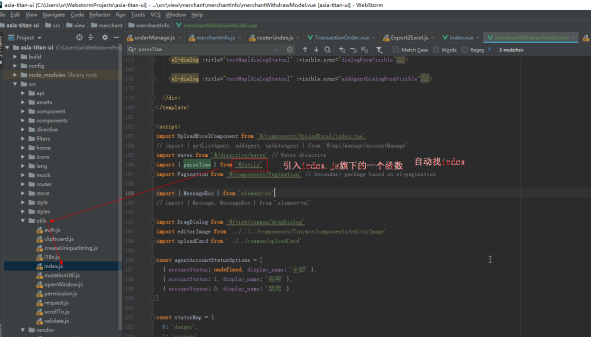


### 封装公共函数

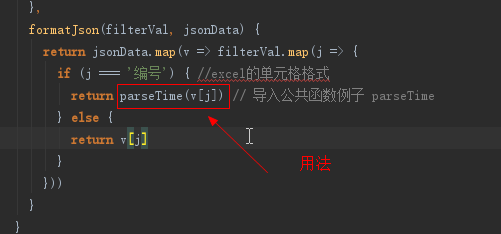
封装：



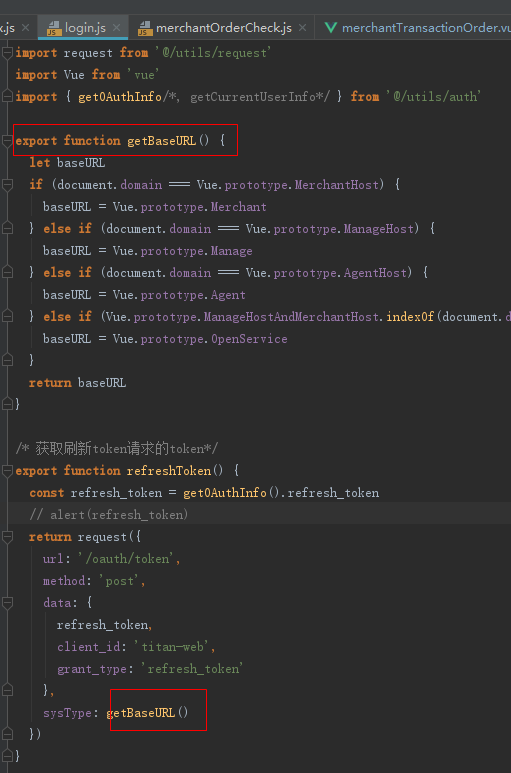
引入：



用法：直接用



其他例子：



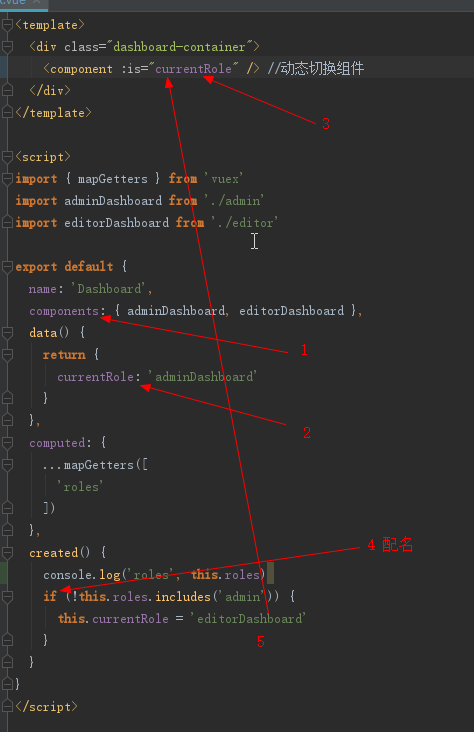
### 8.表格-加行-重置-表格数据提交



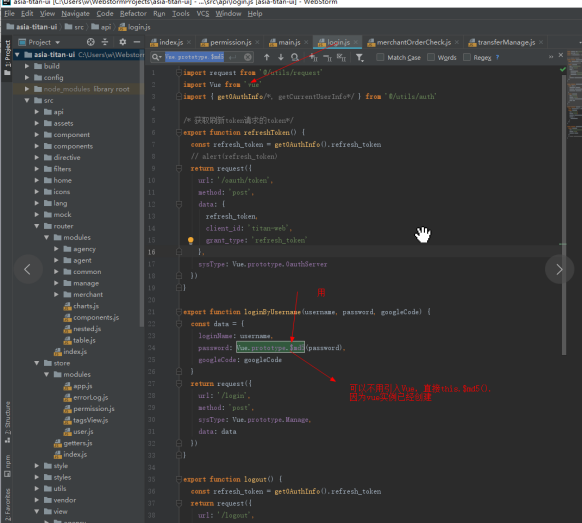
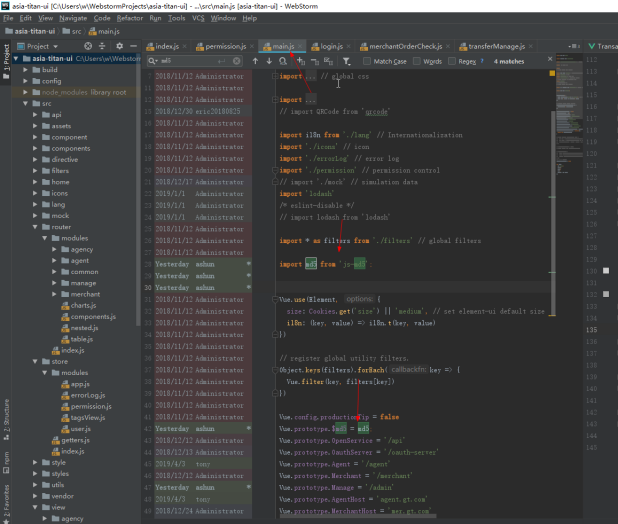
**心得**：Vue-element-admin UI里有删除行功能，不要自己搞

### 动态组件显示

如2个路由，显示其中一个：admin的路由和editor路由



### 存md5等api进vue.propoty



### 10 自定义指令

#### 例子1——按钮点击波纹扩散：v-waves

#### 例子2——权限控制：v-handle=””

### 11 字符串部分隐藏显示

plusXing(str) {

str = str.toString()

const len = str.length - 3 - 3

let xing = ''

for (let i = 0; i < len; i++) {

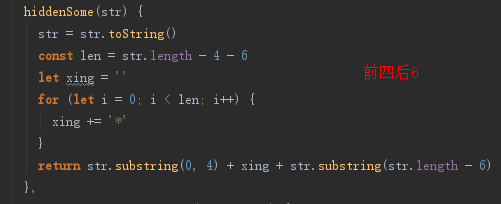
xing += '\*'

}

return str.substring(0, 3) + xing + str.substring(str.length - 3)

}

Tips：3数字可以调（首尾都是显示3位）



## 5. 疑难bug （耗费大量时间解决的坑）

### 1. 循环机制 （涉及引用变量的循环，注意：放在外边还是里面！）

问题：分20个promise导出数据，pageNo全为20？

先后问题：

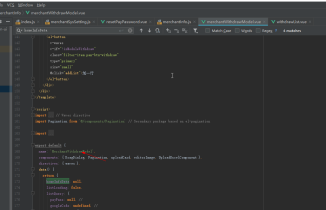
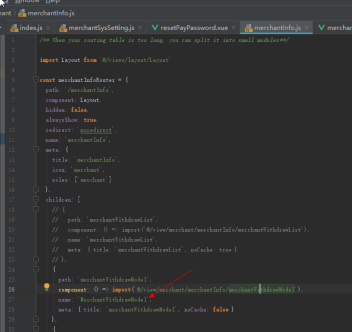
错误顺序：先赋值params，再循环不停替代parmas，最后getOrderList请求引用parmas。所以parmas的pageNo只会是20

解决：先循环在赋值对象。



### 12 页面切换不刷新功能

1. 路由名字和组件导出名字一致



2.

