**Axios从远程读取数据**

**安装Axios**

我们直接使用npm install来进行安装。

npm **install** axios --save

由于axios是需要打包到生产环境中的，所以我们使用–save来进行安装。

**引入Axios**

我们在Pos.vue页面引入Axios，由于使用了npm来进行安装，所以这里不需要填写路径。

**import** axios **from** 'axios'

**服务端拉取常用商品数据**

远端服务器地址：http://jspang.com/DemoApi/oftenGoods.php

（在实际项目中这个后台接口地址是后端程序员提供给你的，你可以随便调用这个接口，我已经放到服务器上了。）

可以先把地址放到地址栏访问一下，是可以正常访问的，并且输出了json格式的字符串，这就是我们需要的远端数据了。这里我们使用Axios的get 方式来获得数据。

created(){

axios.get('http://jspang.com/DemoApi/oftenGoods.php')

.**then**(response=>{

console.log(response);

**this**.oftenGoods=response.data;

})

.**catch**(error=>{

console.log(error);

alert('网络错误，不能访问');

})

},

把axios的方法写到了created钩子函数中，我们使用了get 方法进行拉取数据，如果拉取成功用远端数据对oftenGoods进行赋值。

拉取报错，一般有两种情况：

1. 网络不通：网络状况不是很好，这可以在失败后隔5秒再次请求。
2. 报决绝访问：这种多是后端程序员设置了不允许跨域访问，需要你和后端程序员一起调试解决。

**拉取分类商品数据：**

远端服务器地址：http://jspang.com/DemoApi/typeGoods.php

依然用Get进行拉取，拉取后先用consoe.log(response)查看一下数据结构，让后进行赋值。由于知识跟上边的很像，文字版我就不多描述了，详细可以查看视频教程。

在这里贴出拉取和分配不同分类代码：

//读取分类商品列表

axios.**get**('http://jspang.com/DemoApi/typeGoods.php')

.then(response=>{

console.log(response);

//this.oftenGoods=response.data;

**this**.type0Goods=response.**data**[0];

**this**.type1Goods=response.**data**[1];

**this**.type2Goods=response.**data**[2];

**this**.type3Goods=response.**data**[3];

})

.**catch**(error=>{

console.log(error);

alert('网络错误，不能访问');

})

html模板输出代码：

<ul class='cookList'>

<li v-for="goods in type3Goods">

<span class="foodImg"><img :src="goods.goodsImg" width="100%"></span>

<span class="foodName">{{goods.goodsName}}</span>

<span class="foodPrice">￥{{goods.price}}元</span>

</li>

</ul>

在实际开发中类别也是循环出来的，这里为了教学演示，没有写的那么复杂，你只要明白了如何操作，以后你可以自己增加。就像我这个项目一样，在视频结束后，会慢慢写完善所有功能，最后送给女神，赢得女神芳心。

下节课我们学习订单操作里需要的功能，比如点击商品，添加到左边的订单栏里，增加，删除商品，模拟订单提交到后台。如果下节课一节讲不完，我们就分成两节课来讲。