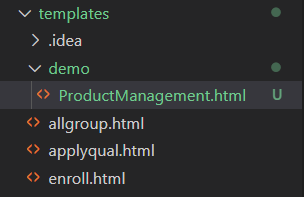
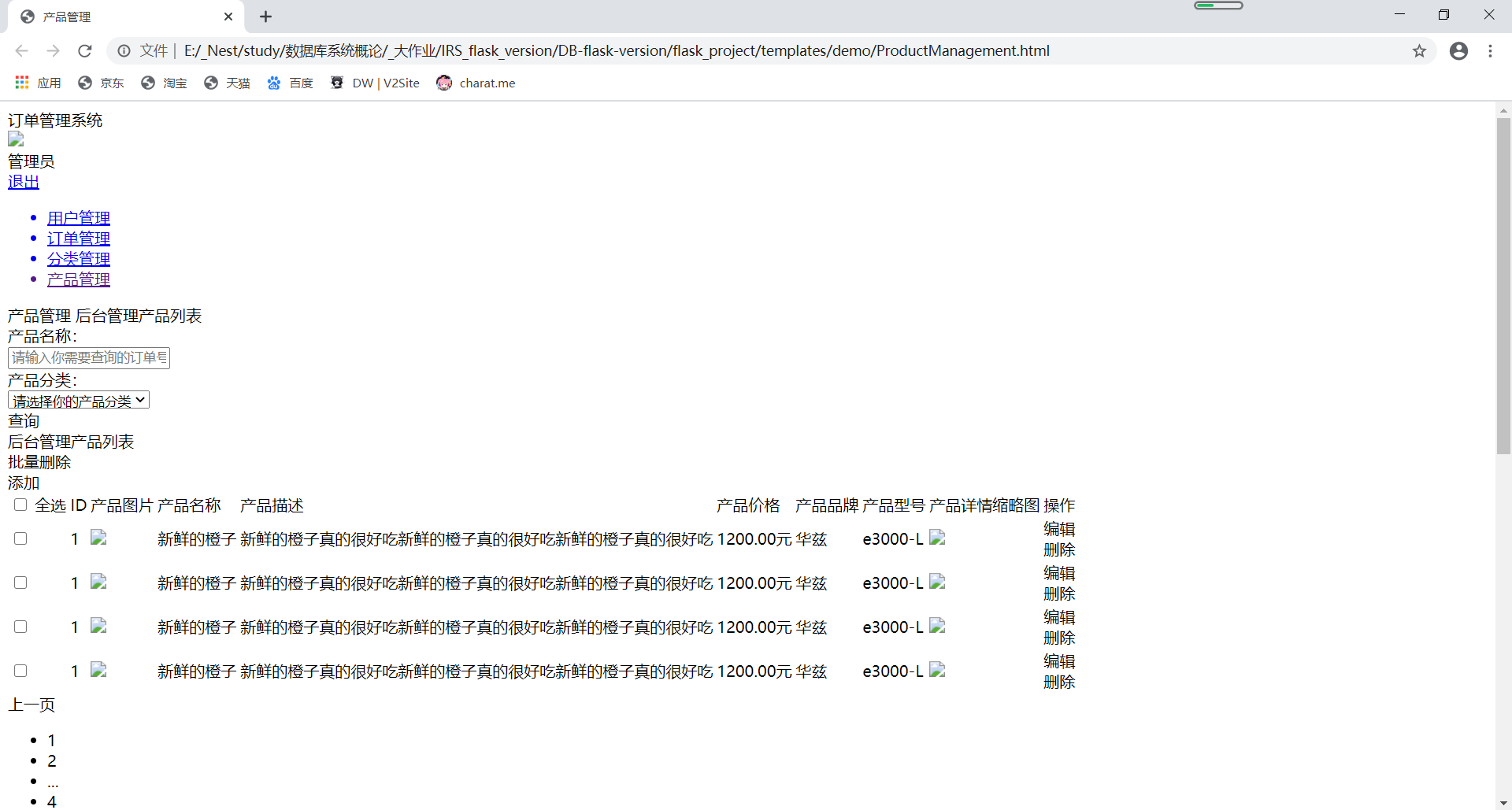
**准备工作**

把html模板放进工程的template文件夹（为了和正式的页面区别，我又在文件夹里新建了一个demo文件夹，再把模板放进去）



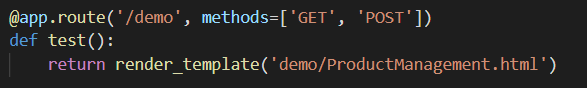
先直接点开模板看一眼



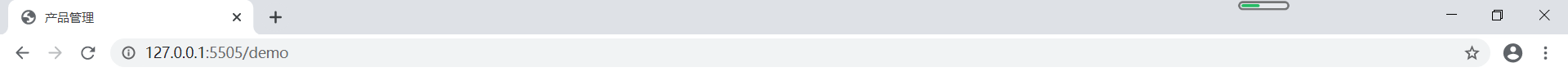
好的，有内容，但没得样式，那就是html中引入css文件的链接不对

这个时候还没法通过url来到达这个页面，因为没有在app.py里面添加这个页面相应的路由

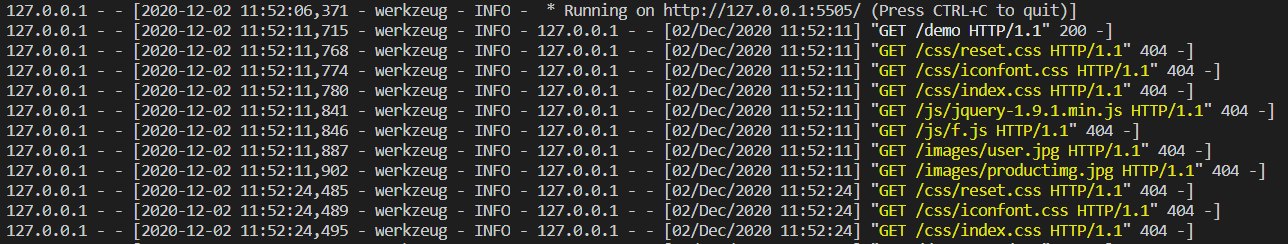
先添加路由



然后把app.py运行起来，就可以通过这个域名访问到模板页面了

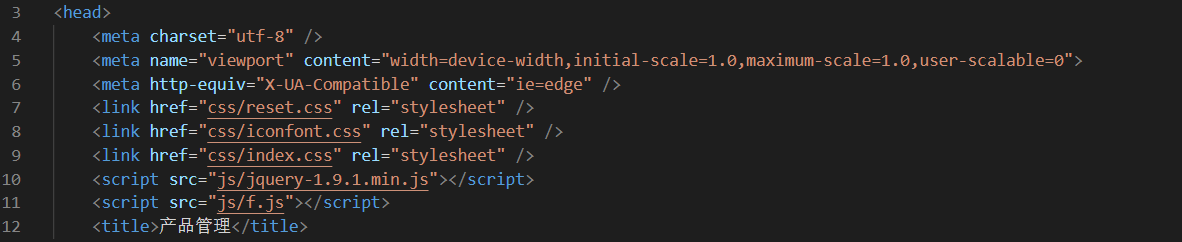


查看终端输出，发现html试图链接引入的文件时，都是404未找到，看得出来是链接路径不对

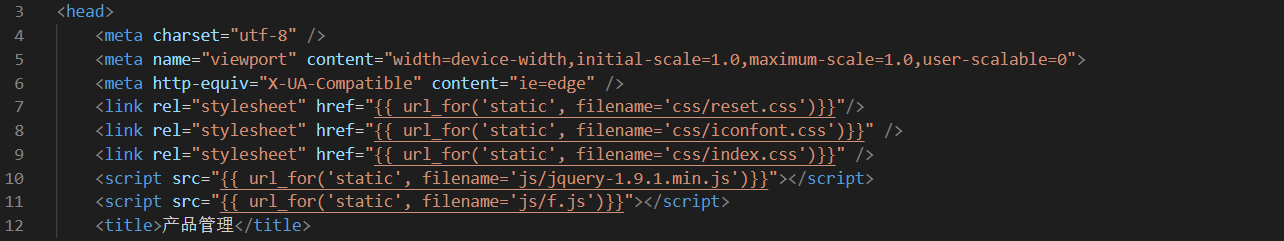


修改模板html的引入链接

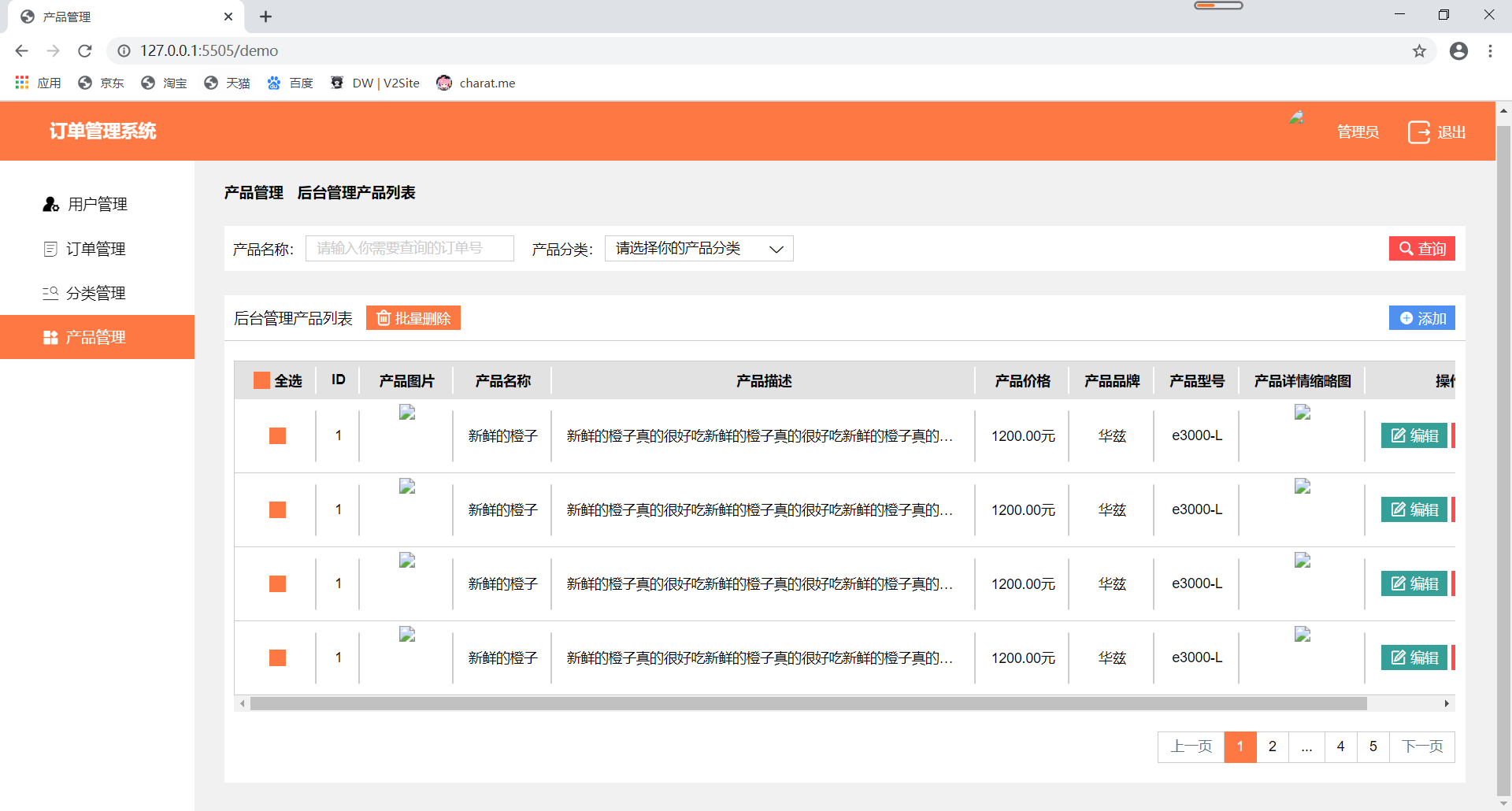
前：



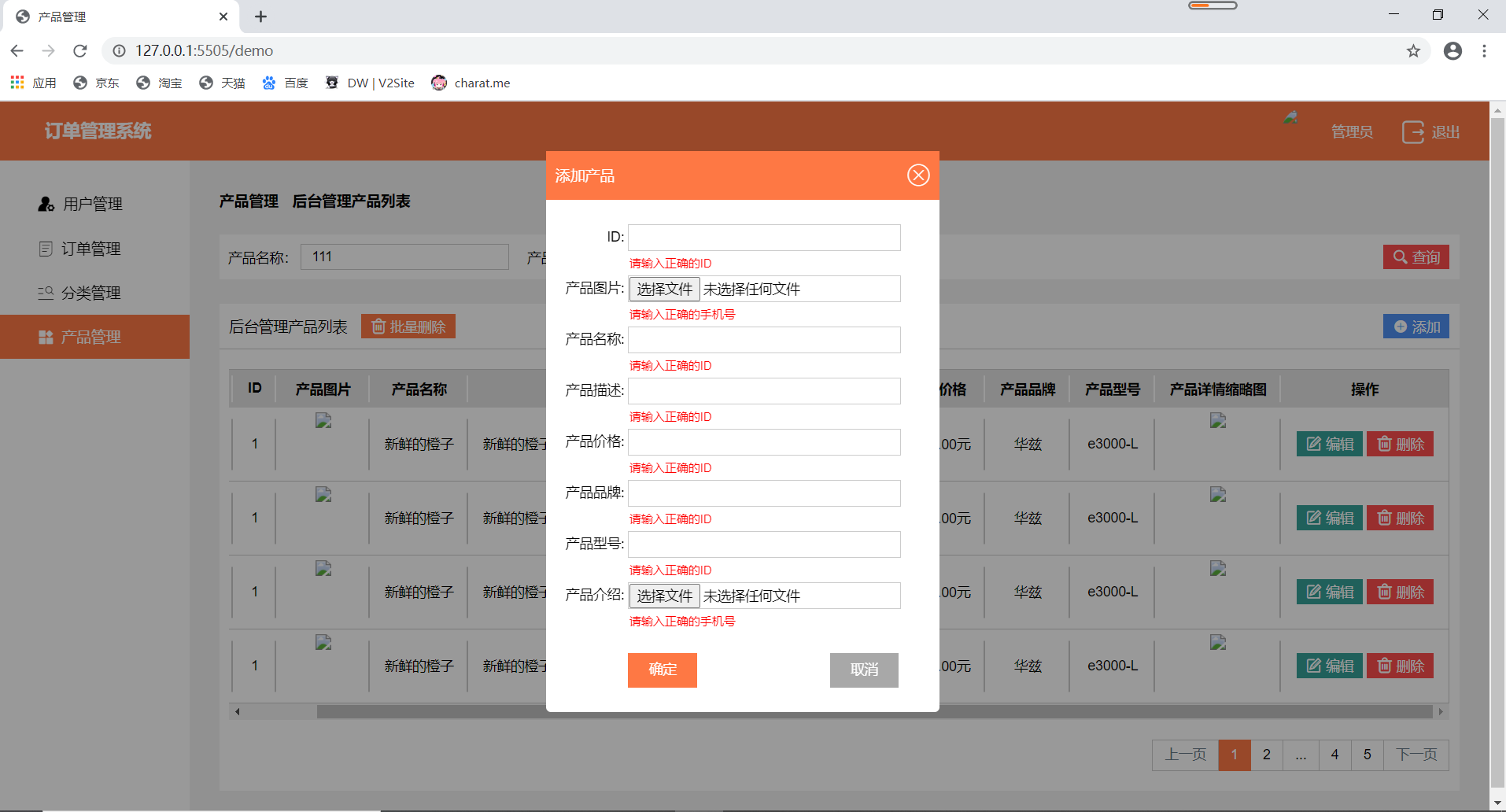
后：（我就从你们已经修改好的html里面复制过来啦）



现在虽然还是有一堆链接是404，但至少基本样式是成功引进来了



四处点一点戳一戳，发现点添加、编辑和删除能正常弹出弹窗填写信息，但没有实际的提交等功能，其他按钮也没有实现功能，说明负责这些功能的js文件没有成功引入或模板没有包含

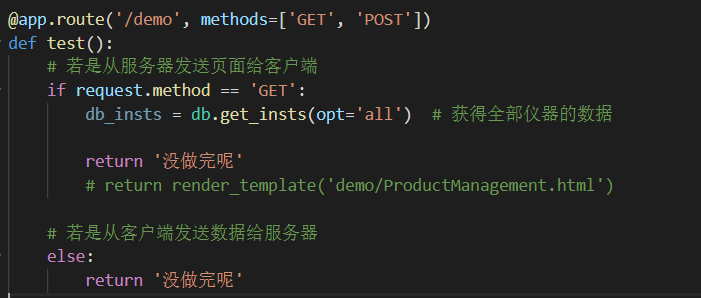


**检查后端传回的数据**

总之我们先把这个页面改造下，让仪器信息能够显示出来

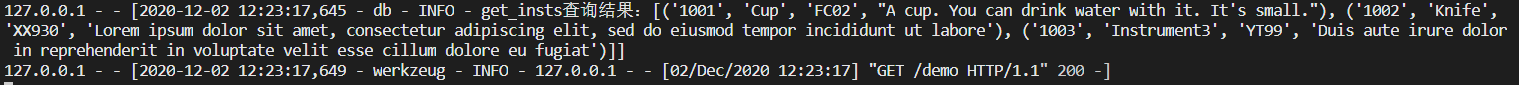
改一下app.py，首先看看后端传给前端的仪器数据是怎么样的

（这里我修改了一下get\_insts函数）



运行起app.py，进入网页，查看终端输出（因为我在get\_insts函数里写了输出，你要是想在其他地方看输出，就在对应的地方写个输出语句）

你就能看到输出了我瞎编的数据



对照一下数据库，是正确的，get\_insts函数返回的值是一个元组组成的列表，每个元组的元素是一个属性值，这个列表现在储存在变量db\_insts里

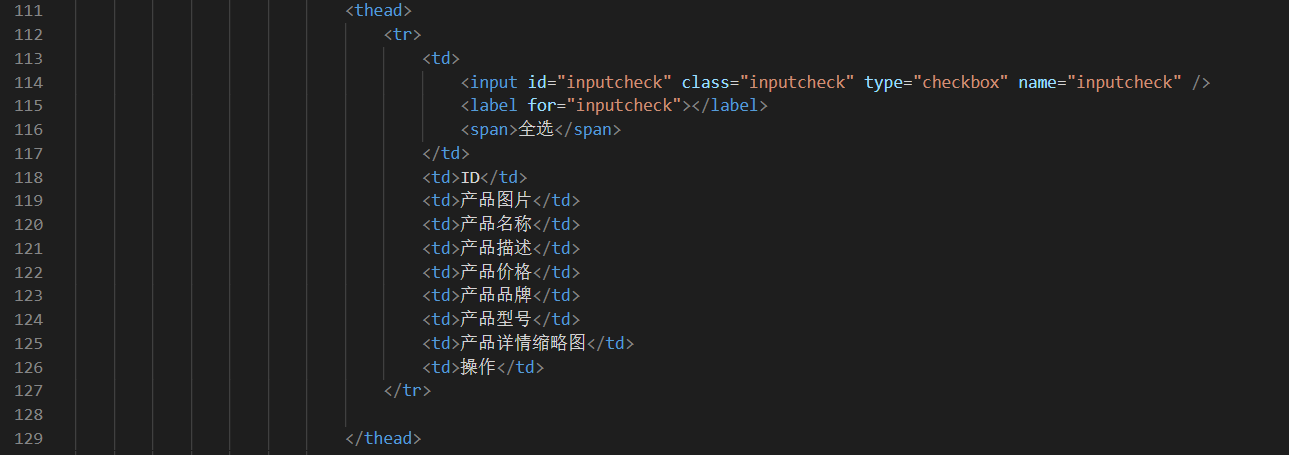
**修改前端html使仪器信息正常显示**

然后开始看前端代码，可以直接看代码文件，也可以在浏览器里按F12打开开发者工具看，这样的好处就是你能很直观地看到哪块代码对应哪块区域，但是这里显示的代码可能会和代码文件本身有差异（比如代码文件中的变量会直接变成字符串）

当前的目标只是让仪器信息正常显示，所以先不管别的，直接看表格的部分，也就是从这里往下的一大块



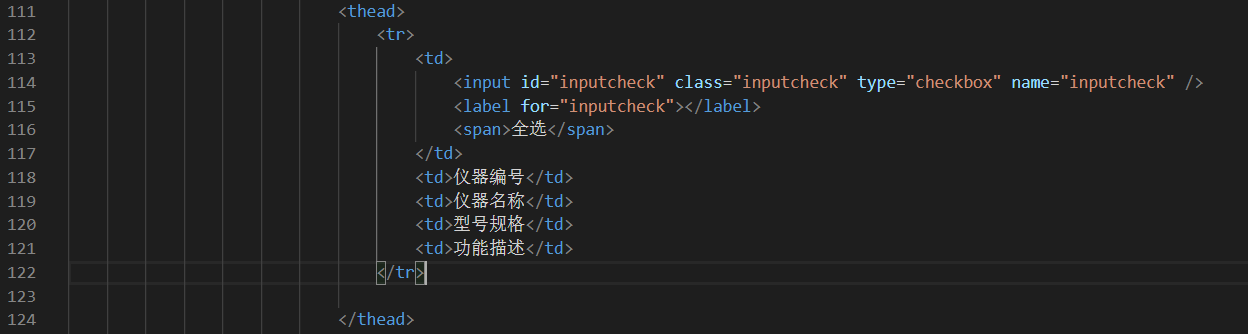
对照网页或者通过标签(<thead>)可以知道，这一块就是对应表头的





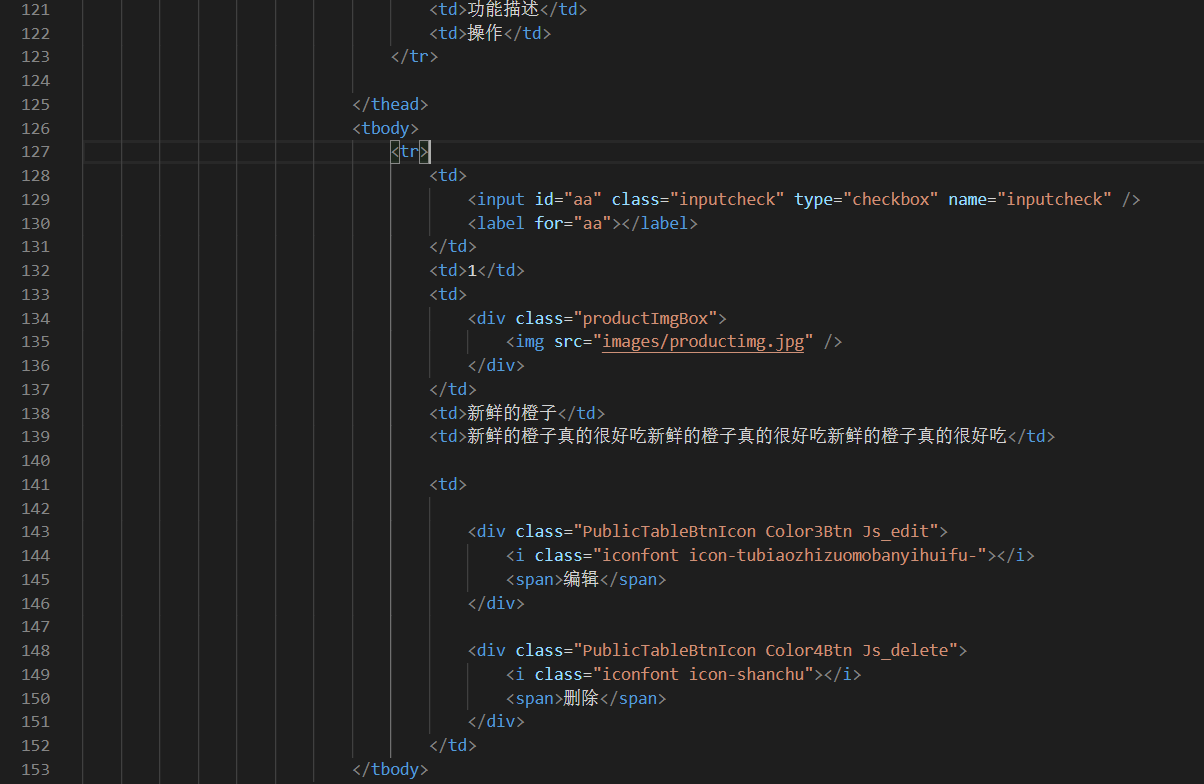
这里要改的话，可以直接把里面的字改成仪器属性名（因为要显示哪些属性我们是事先知道的），也可以从后端把属性名传过来，然后用for循环遍历，效果是一样的

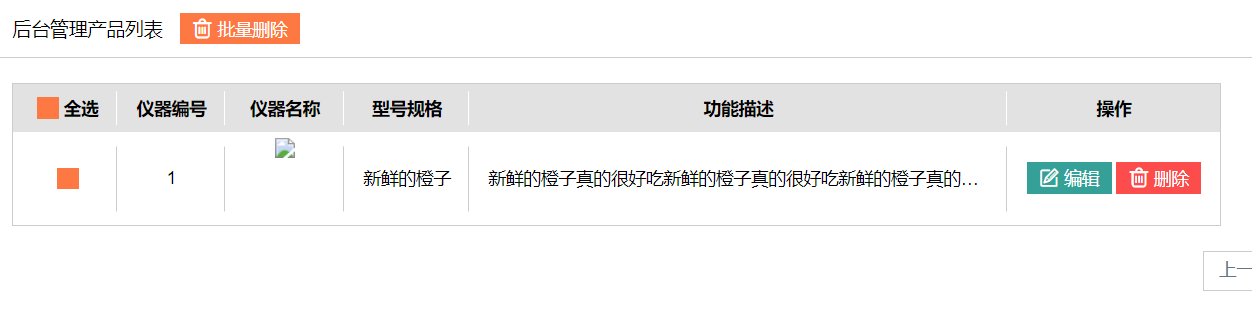
因为后端现在没有传属性名过来，我们就先直接改字





下面的表项内容也是我们要替换的，为了方便我们先把模板删到只剩一行表项，然后把不要的列也删掉



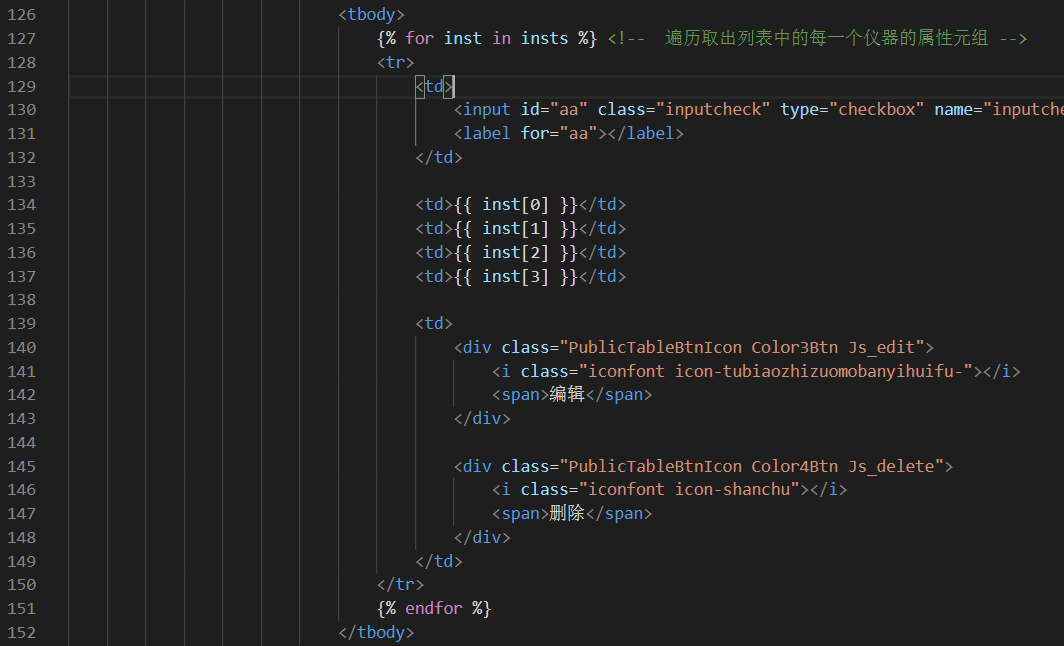


这样就可以用for循环把这一行表项的代码括起来，把内容的位置换成变量了

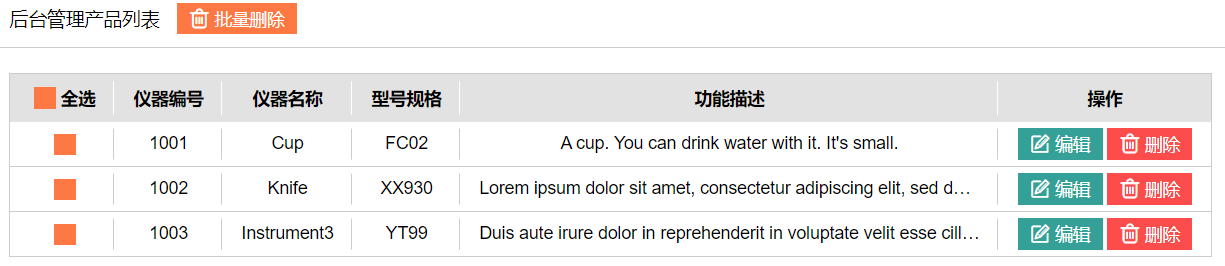
首先回到路由，把后端的数据传到前端



然后在前端加入变量和for循环



显示就正常嘞



但是还有个小问题就是功能描述显示不全……这应该是要改CSS了

而且页数并没有起作用，这些都需要后续修改

**给编辑按钮实现功能**

Html文档是一棵树，每一个元素都是一个节点，所以是可以通过父子关系从一个节点到达另一个节点的（详情请看html dom）

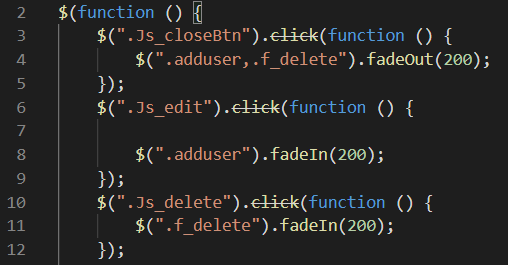
因为每一行都有一个编辑按钮，所以对每一个按钮，我们可以通过父子关系以及id定位到具体的这一行的表格

先给每一格加上一个id



然后虽然html里面有弹窗的内容，但看了老半天html也没有发现【点击按钮弹出弹窗】的关联，所以应该是写在了引入的js文件里，从一开始的引入文件里找到f.js

在这里



首先点开弹窗的时候，我们要**把表格中的仪器信息填进相应的输入框位置**，那么我们也需要给每个输入框单独的id

回到html改改输入框内容



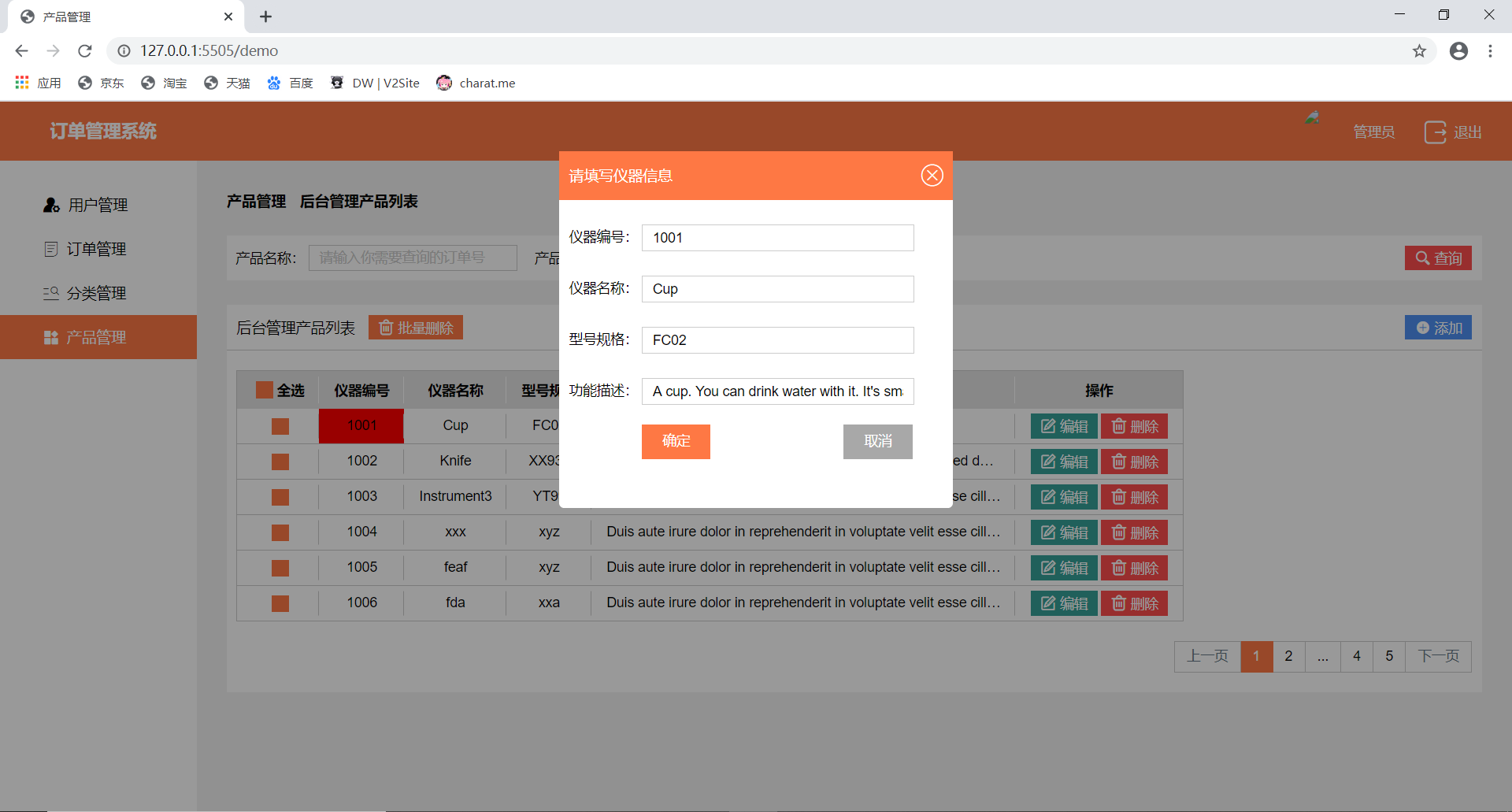


然后让用户点开弹窗时，取该行的数据填充输入框

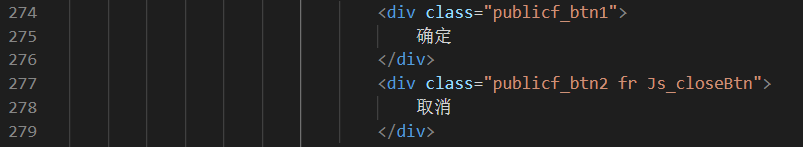


效果出来了！好耶！虽然还有可以改善的地方，比如功能描述的输入框可以更大点，不过这个可以之后再说

如果没有效果的话，也有可能是浏览器缓存没有清，可以用ctrl+F5强制刷新浏览器试试



接下来要让用户在弹窗里修改好信息后，**点击确定按钮，把输入框里的数据提交给后端以更新数据库**



这确定键甚至连个id都没有，给它补一个

但这里有一个坑，就是这个模板中点击“添加”和“编辑”跳出来的是同一个弹窗，你按下确定的时候，需要判断一下操作到底是添加还是编辑

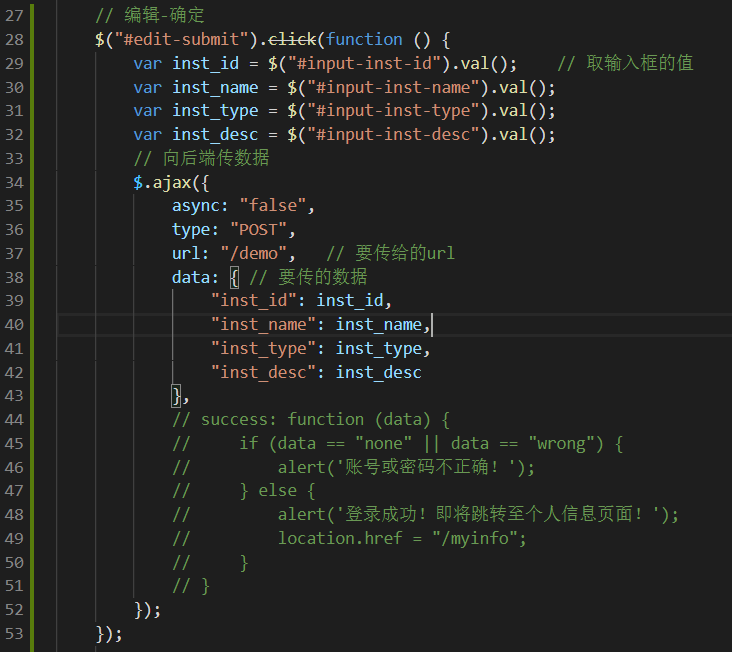
这个判断可以先把数据传回给后端，由后端查询一下仪器ID是否在数据库中有，若有，则更新；若没有，则插入

也可以在前端让添加和编辑分别跳出不同的弹窗

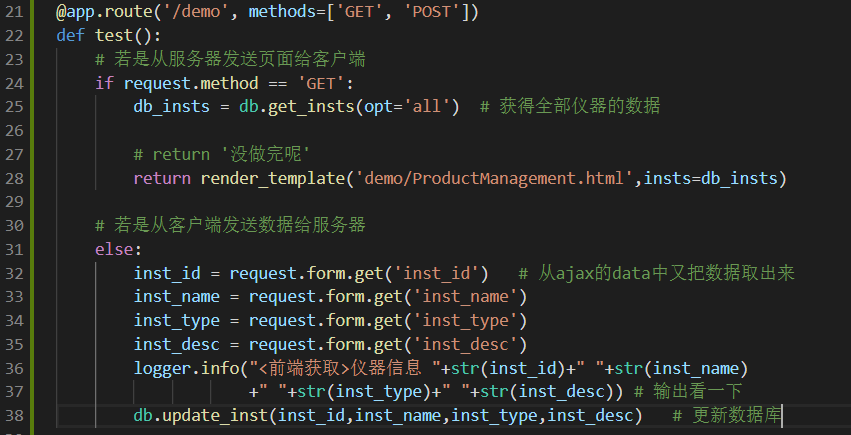
我们就先默认只要按了确认，就是编辑某个已经存在的仪器的信息



然后写发回数据的js代码



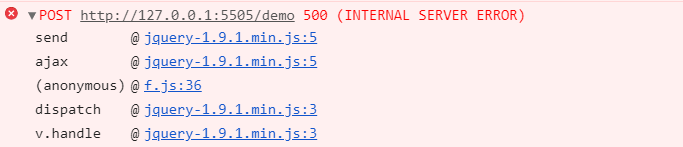
回到app.py写接收数据并处理数据库的代码



又回到f.js写从后端返回前端后要执行的代码

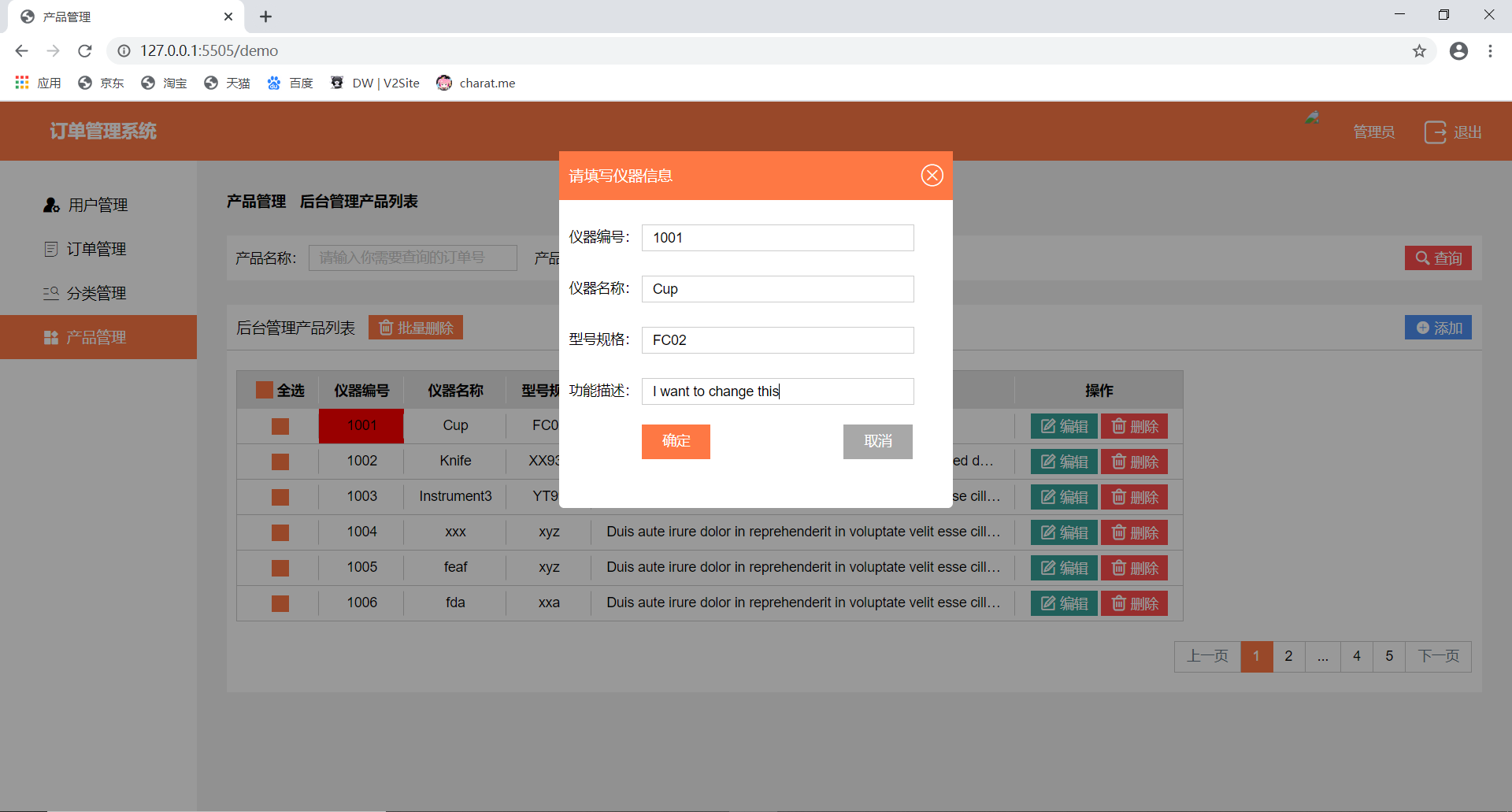


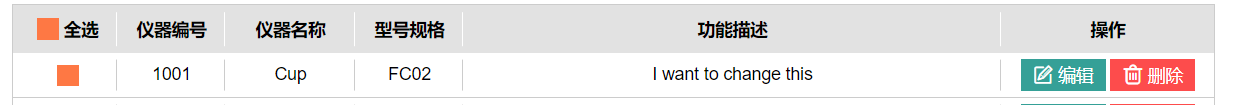
点确定按钮，发现没反应……点开开发者工具一看，错误代码500，呃……可能ajax哪里写错了……



总之一番折腾，把html文件名改成了demo，移到templates目录下，又行了，可能就是url不太对

总算行了，我本来以为不用花很多时间的……





以上，过程有很多地方我也是边查边做，也有遗留问题没解决（比如如果修改了仪器编号，sql可能会出错，因为是主码，所以最好别让用户能改）

加油！