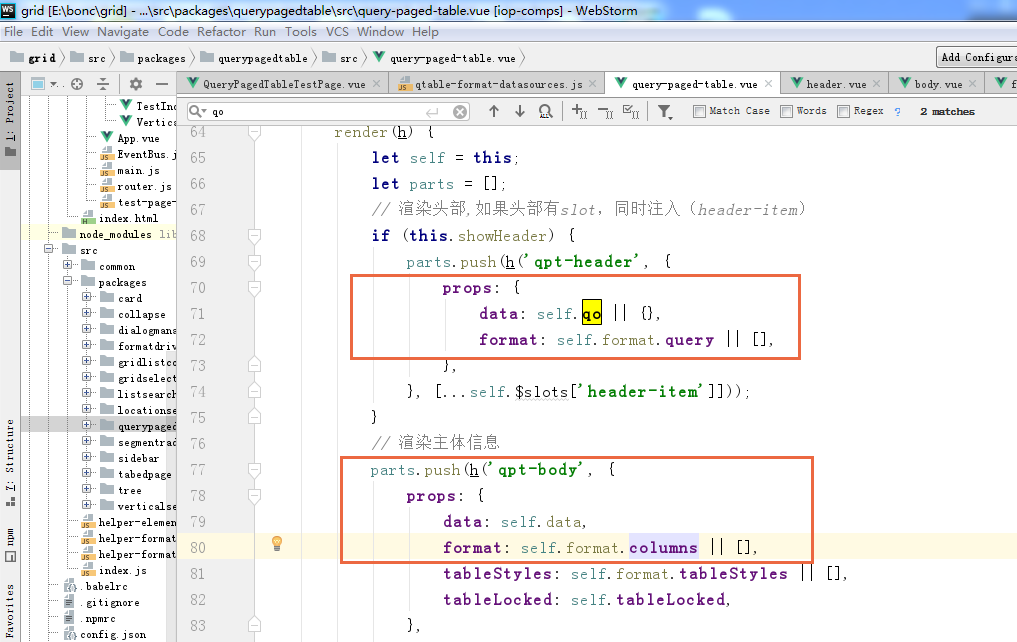
rebuild-fornted的关键点

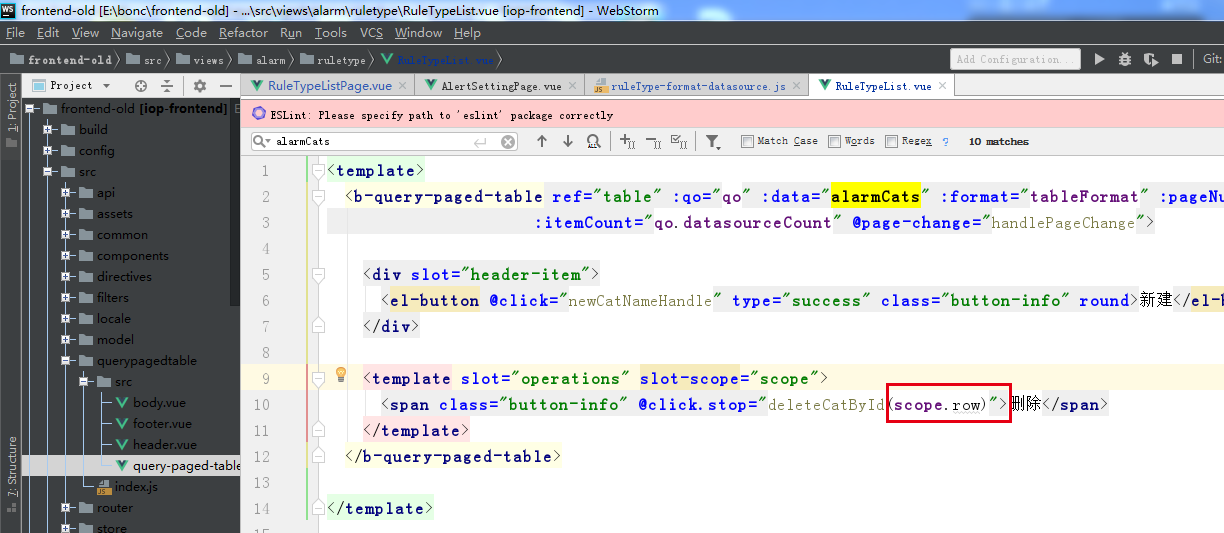
1.随时改变数值的，表示头和尾的都写在qo里，比如需要v-model的Input控件，就写在qo里，这样可以实现双向绑定

2.看这2处就知道该把数据放哪个位置了



3.问题：记得之前写组件时不能自发地从props传变化到子组件，为啥在iop-comps里又可以了？记得好好研究一下

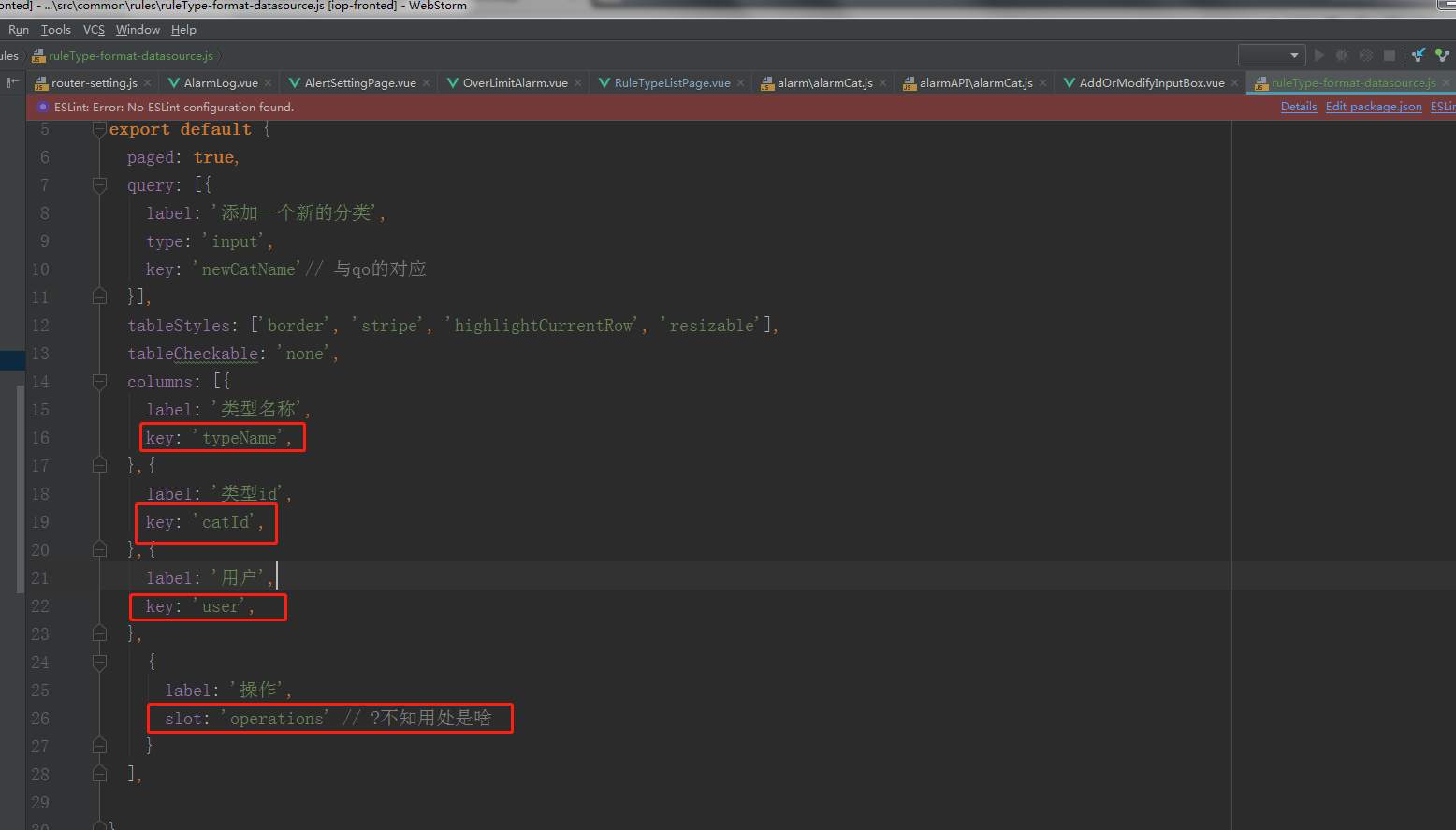
4.scope还是不会用

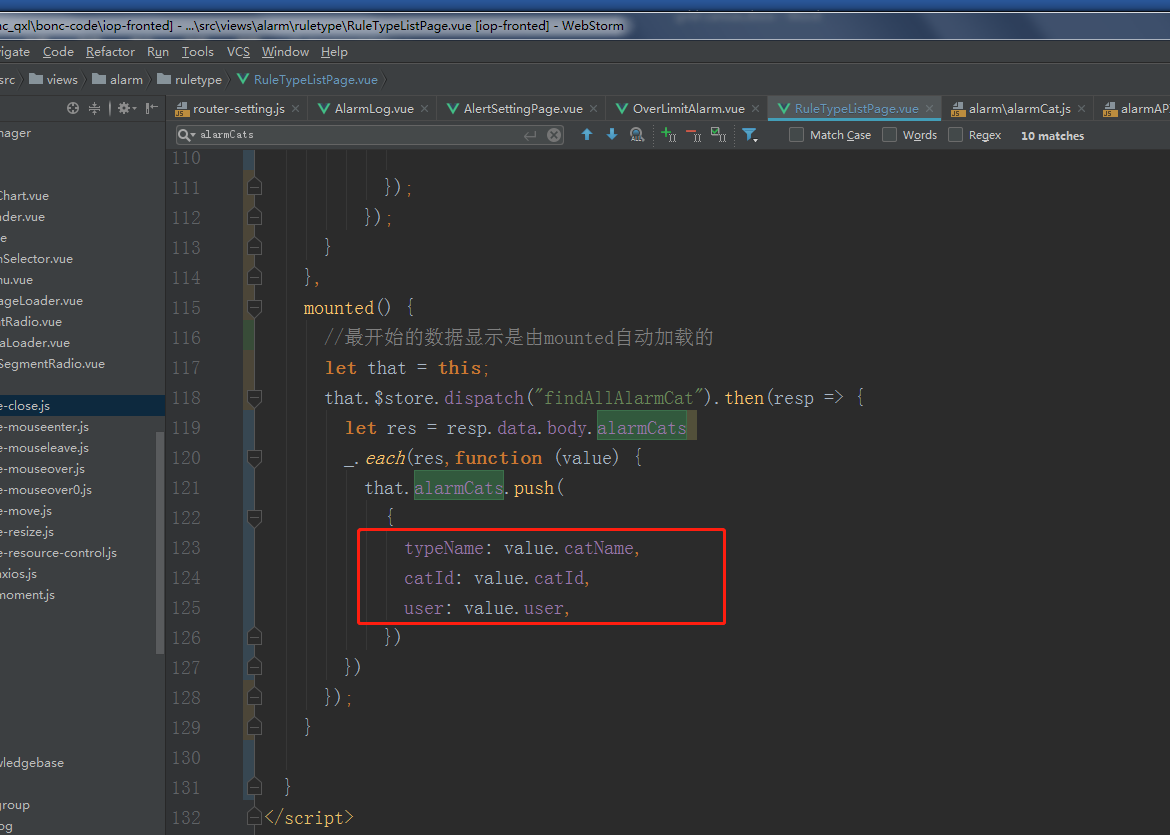


5.好好看看callback的



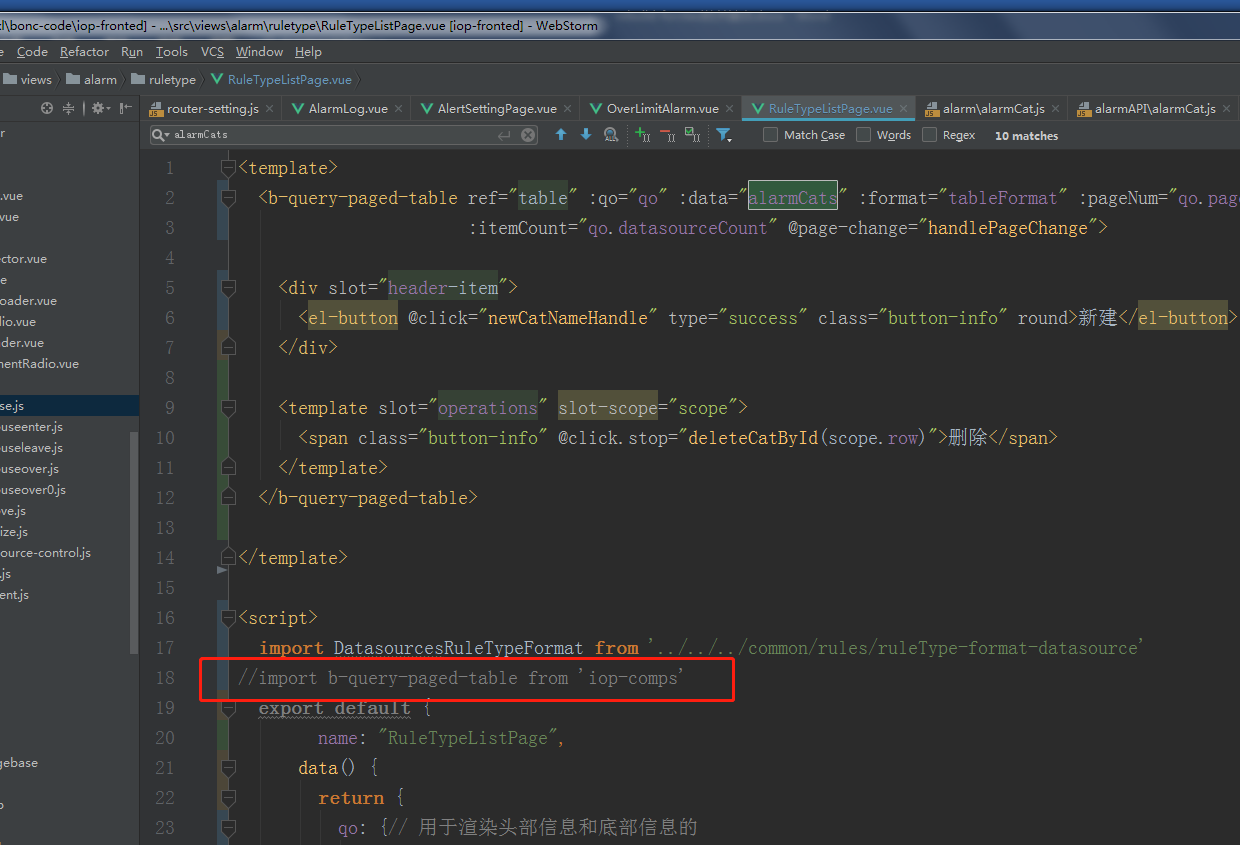
1. 这2个地方要对应：

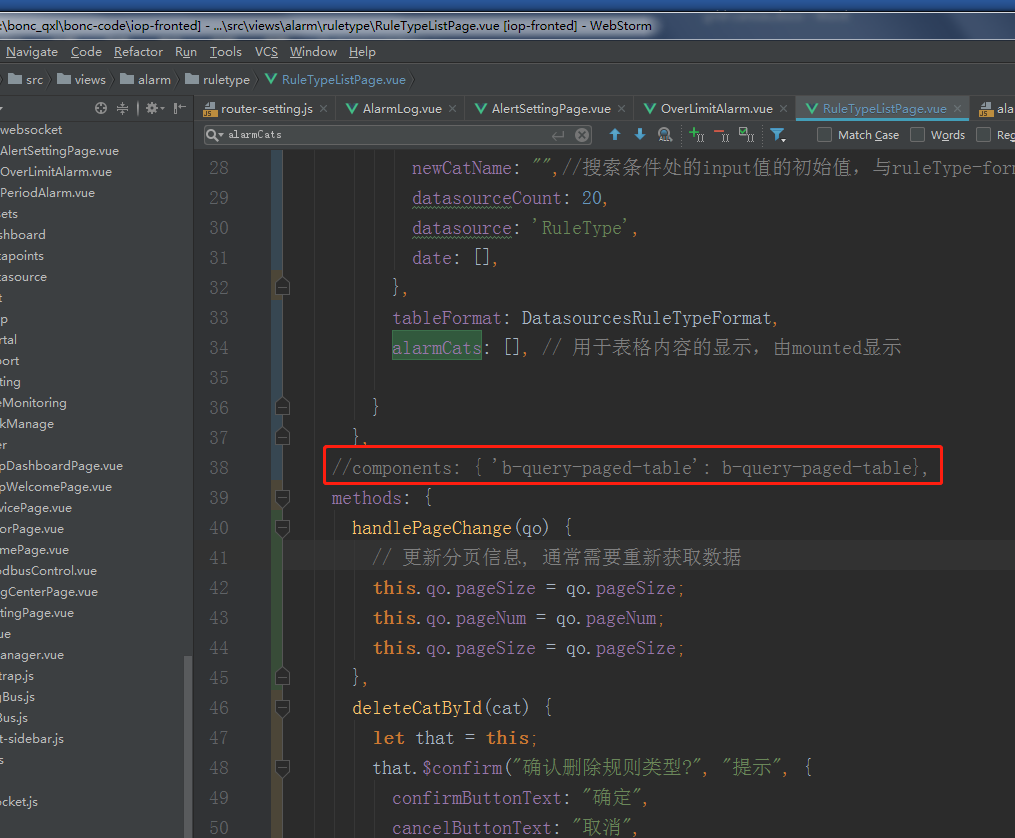




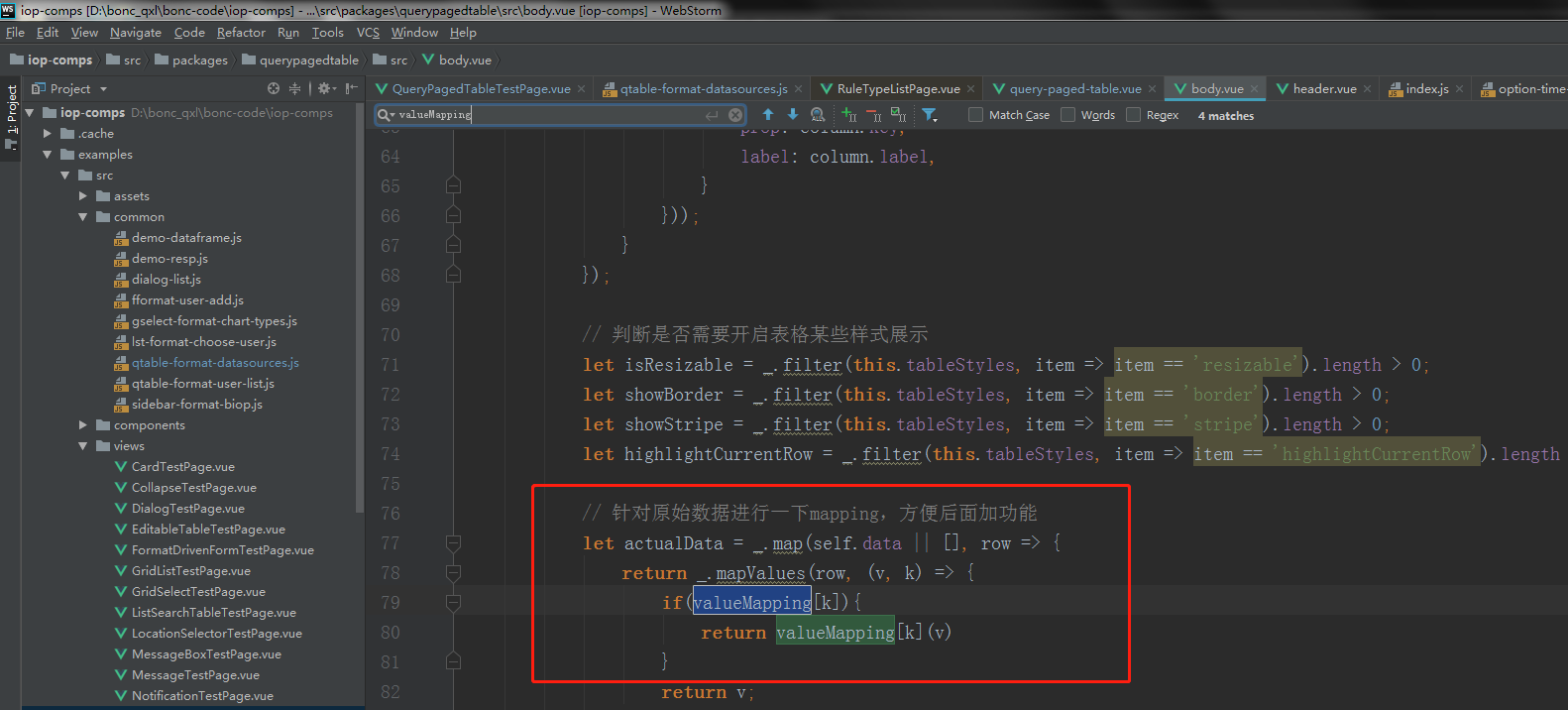
且后者的文件中所有用到这个的地方都要这么改

1. 不需要引入组件，那组件是在哪里引入的？



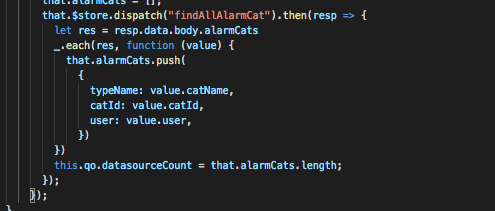


问题：这个不太懂怎么遍历的



代码规范：

1.



代码规范：

date为什么是个数组？如果是时间范围，变量名用dateRange

最好用self，统一一下

缺注释

顺序 created -> mounted -> methods

获取全部通过绑定一个事件，需要重新获取，emit一个事件

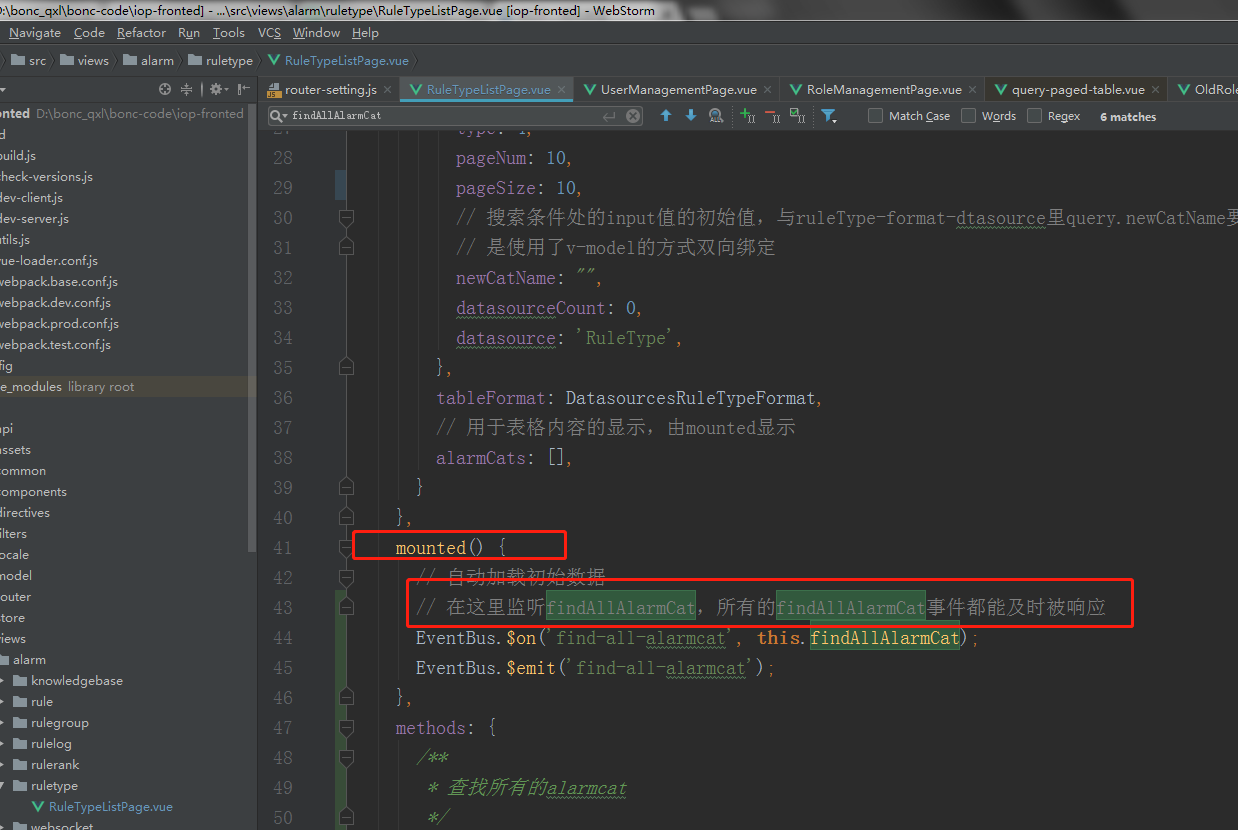
新的规则不是弹窗么？校验在弹窗里面进行

方法命名，要是名词

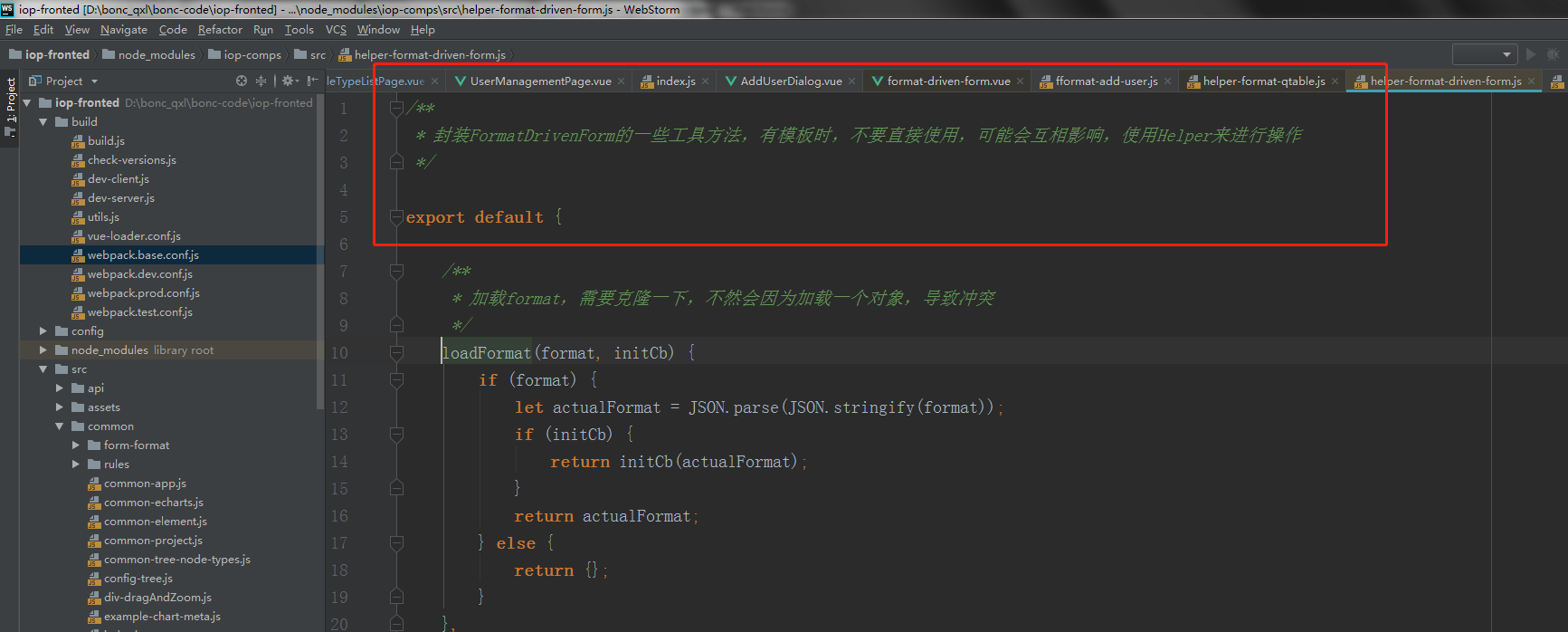
取消删除不用提示

这个用@的绝对路径

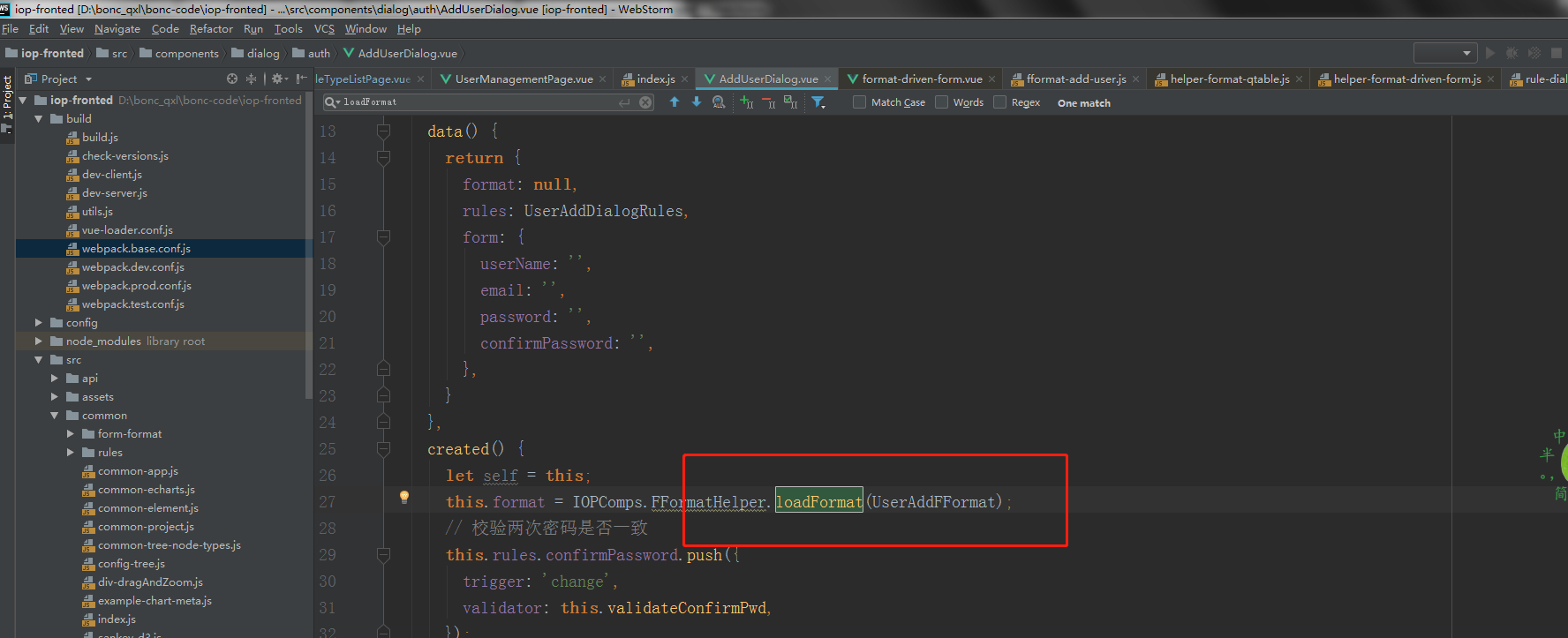
2.



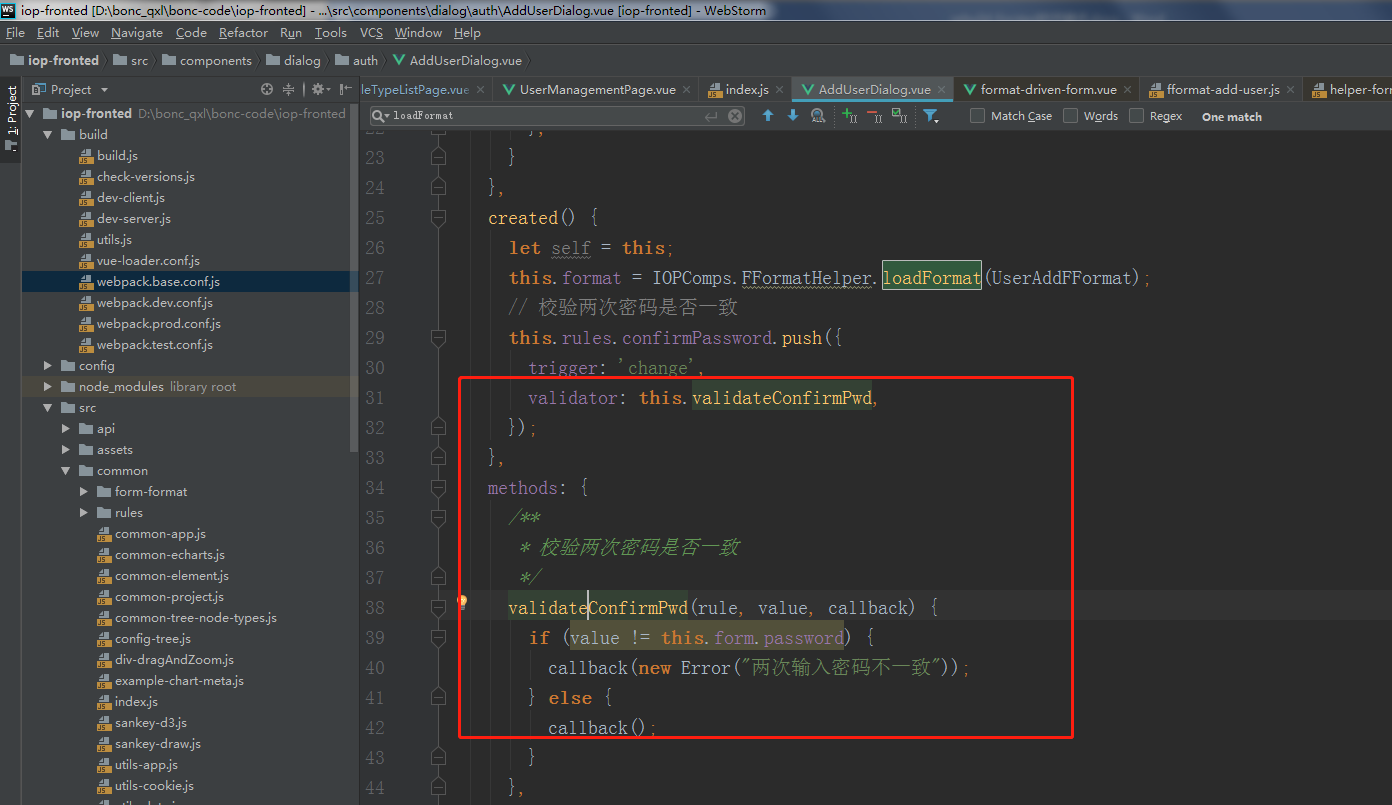
1. 问题：为啥要这样操作？



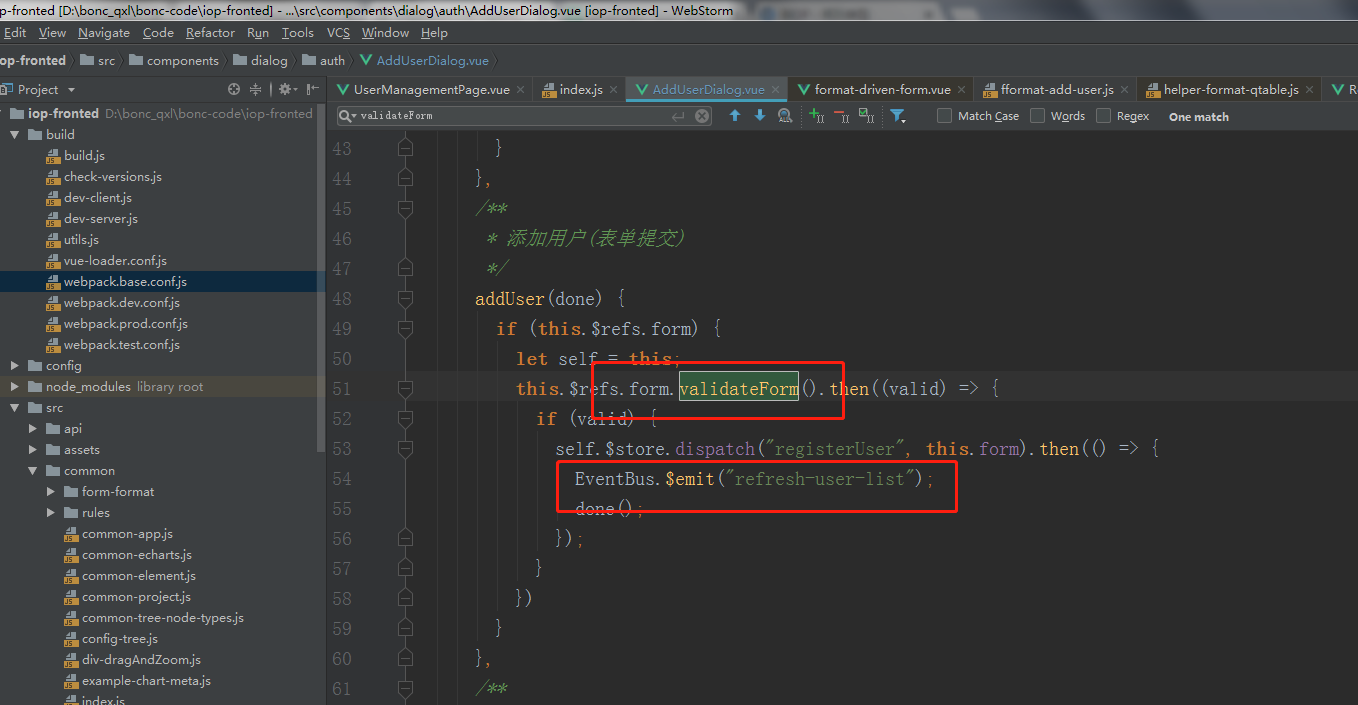
1. 有2个文件都有这个函数调用的是哪个？



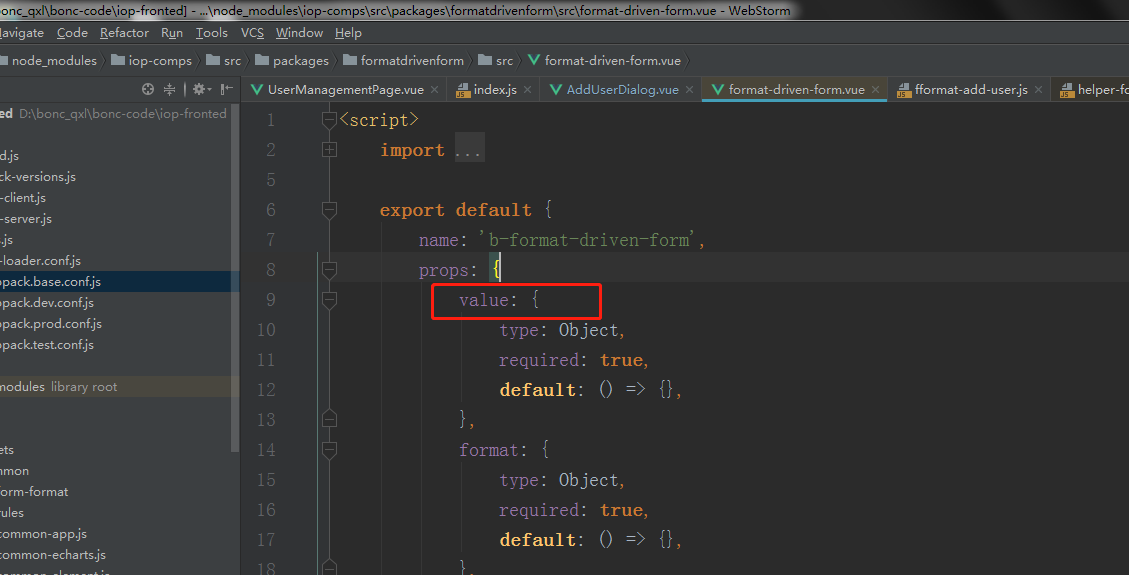
1. 这种参数都没有传过去的也是可以的吗？

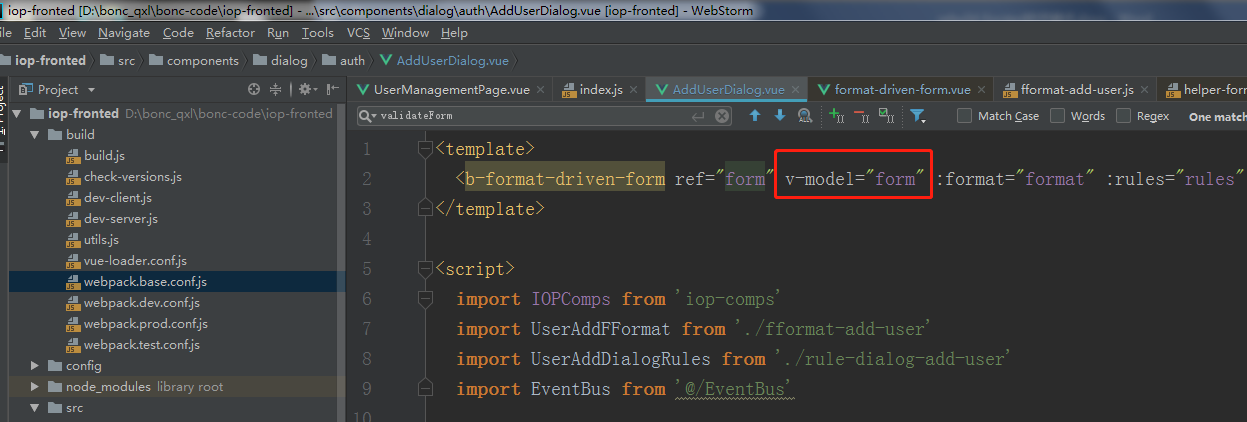


1. 跨域了好几个文件：

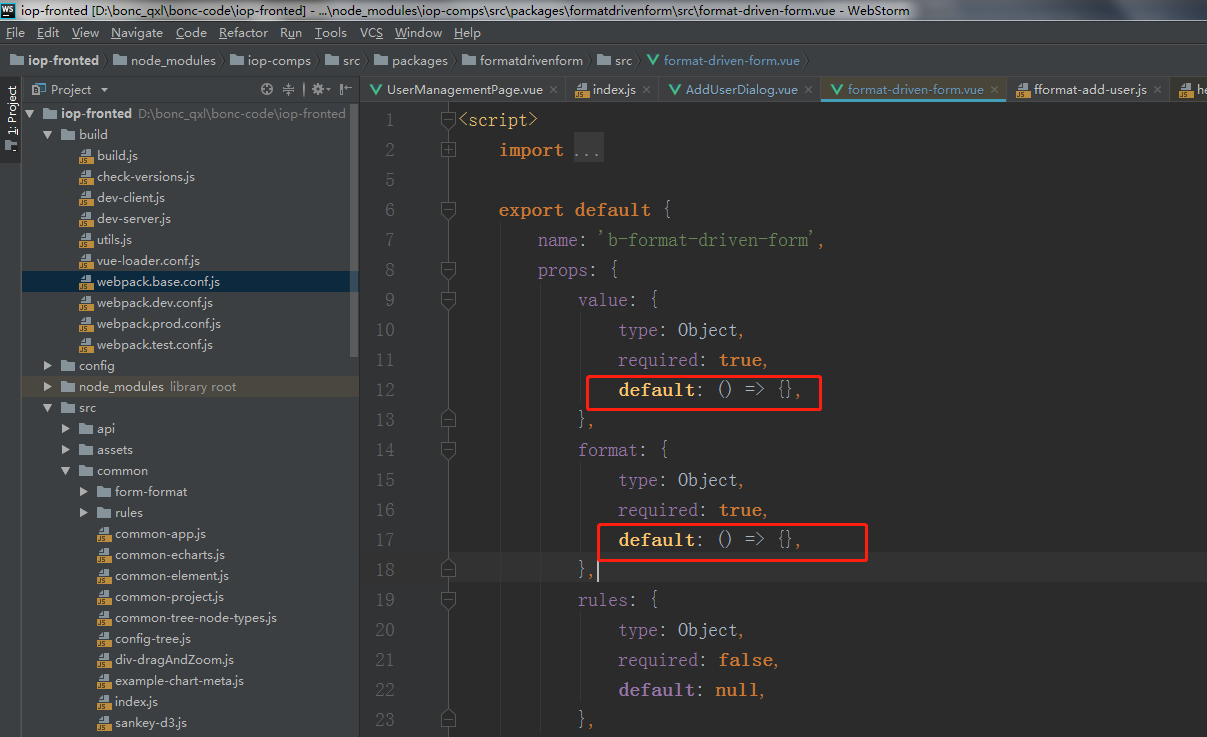


7、这个value即是那边的v-model:

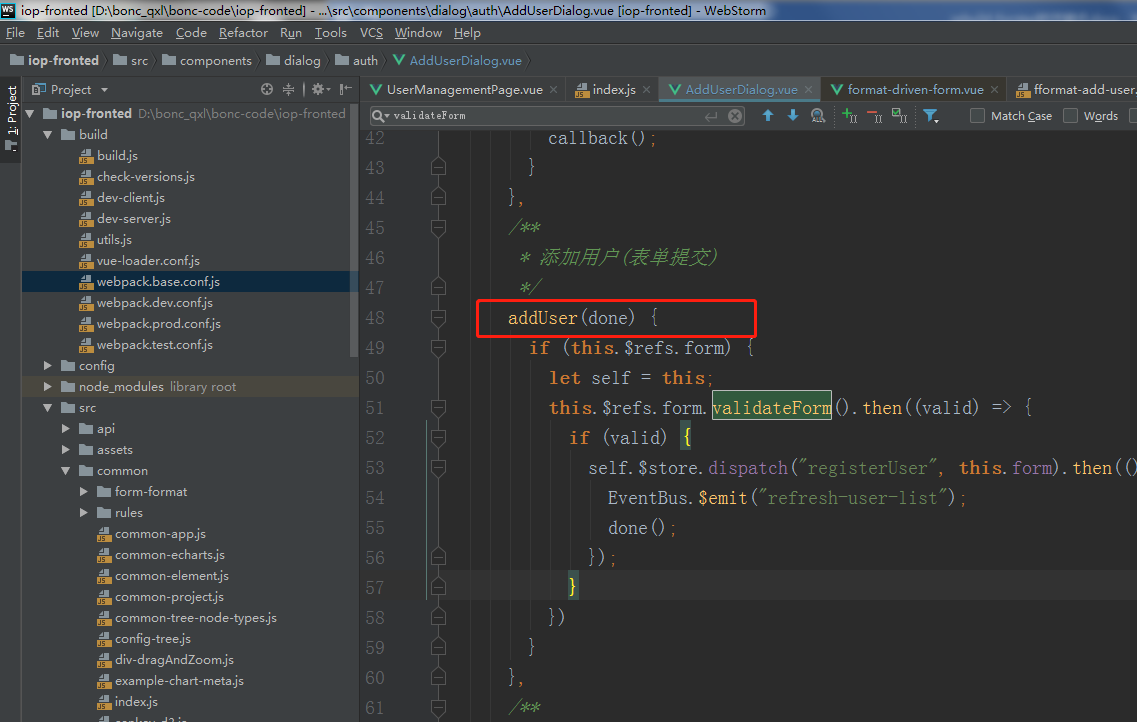




9.default的作用是啥？

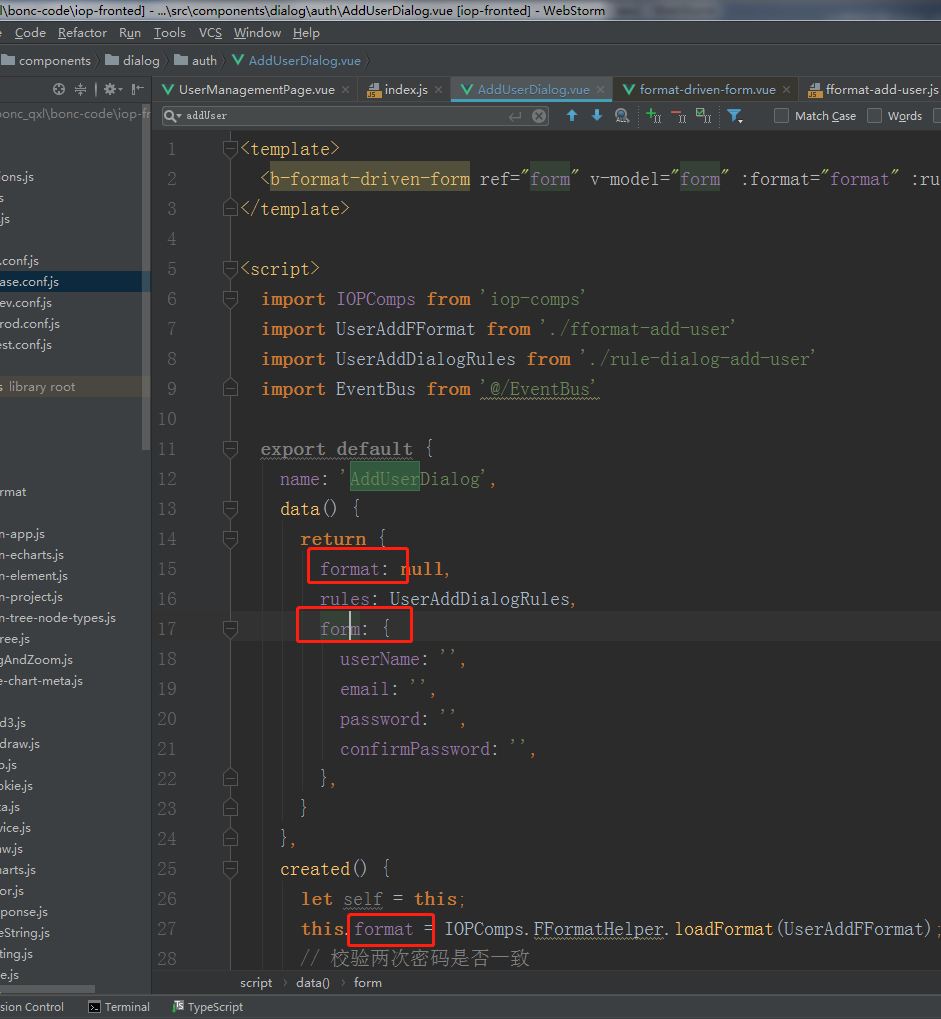


不太懂为啥为跨越这么多个文件执行

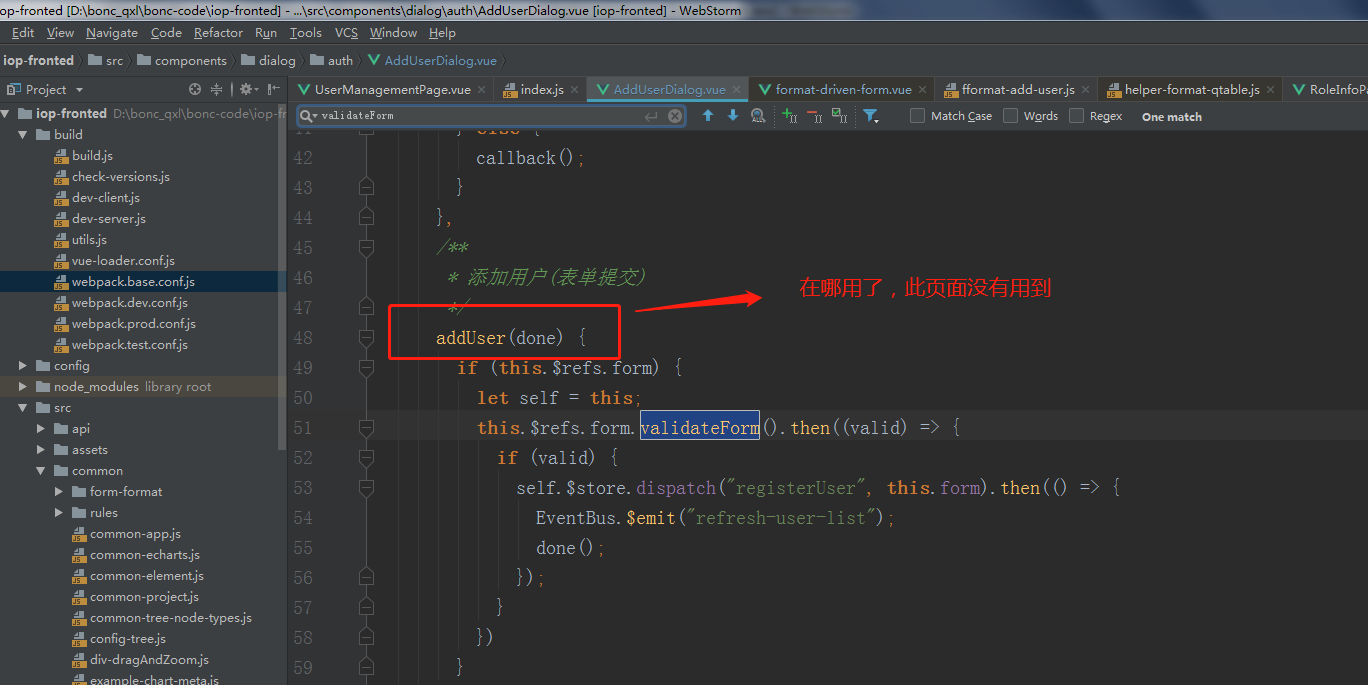


2.dialog

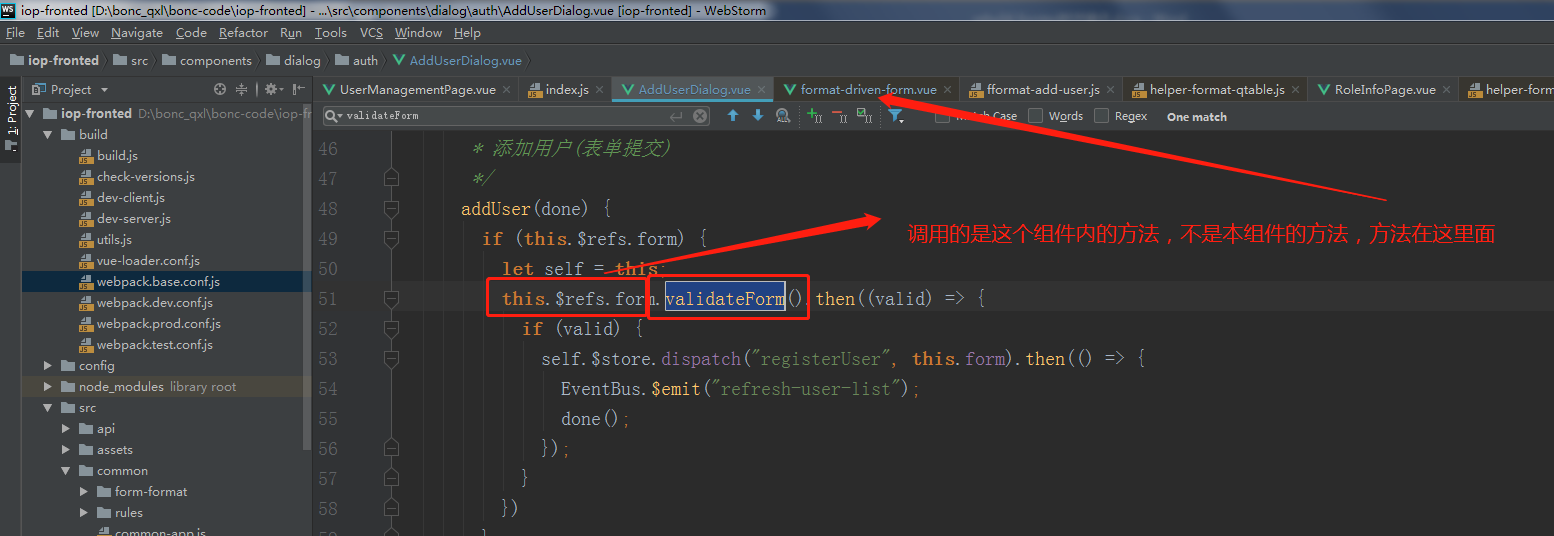
1.form和format的区别在于，前者是后者的软部分



2.

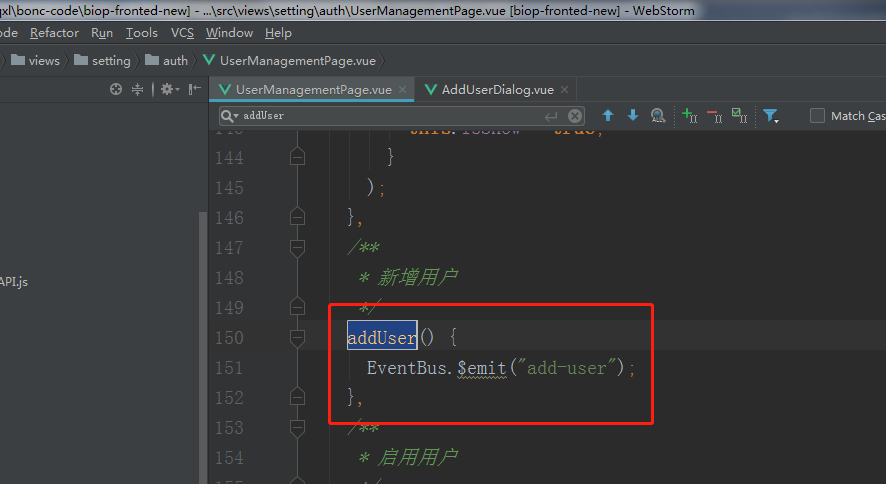


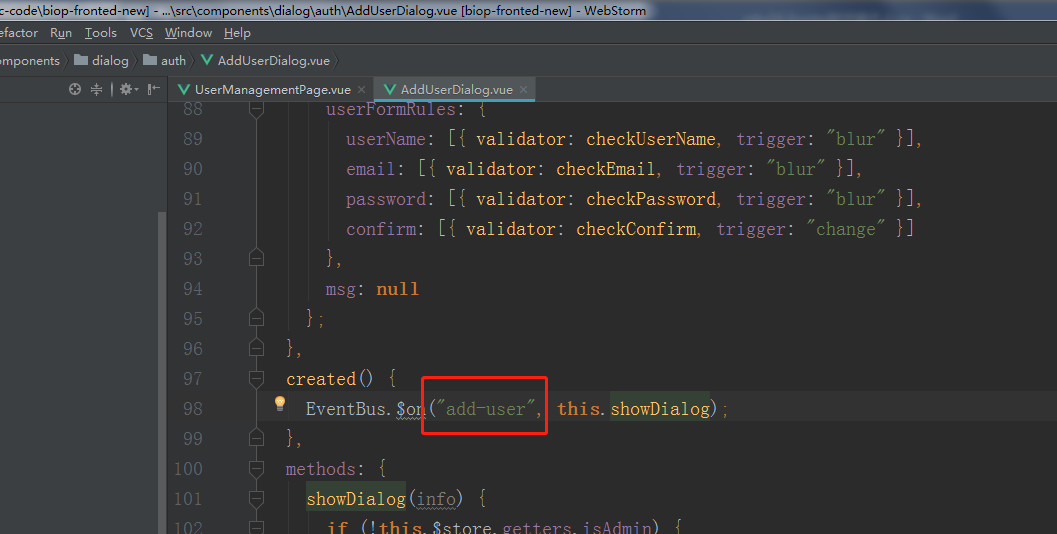
3.



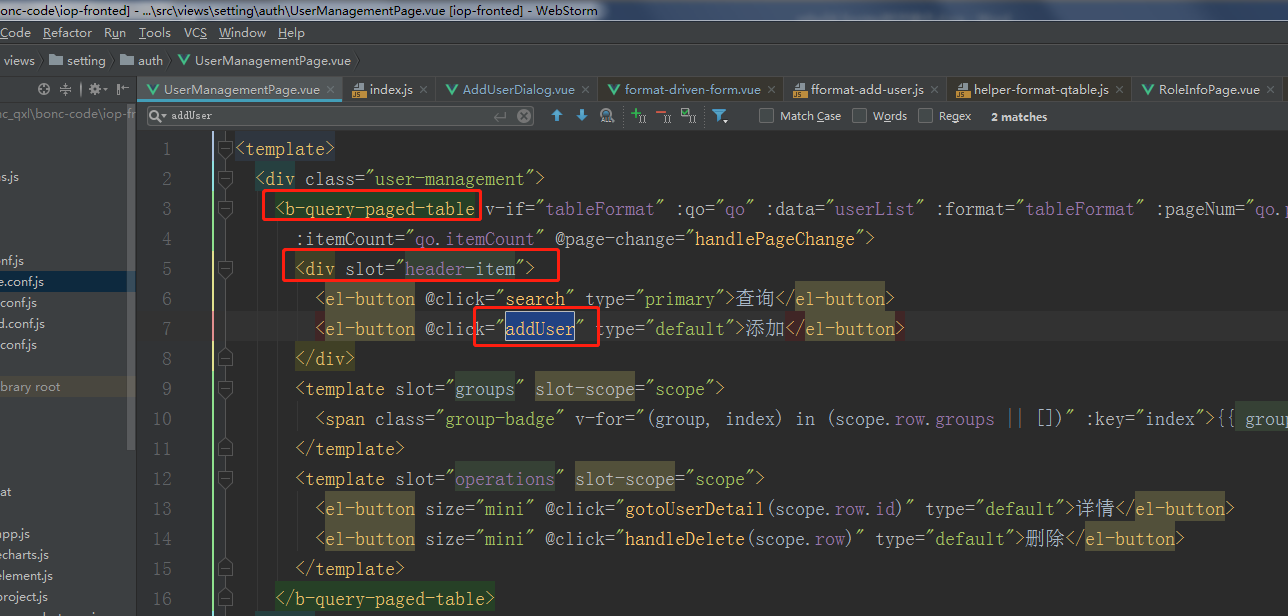
4. addUser

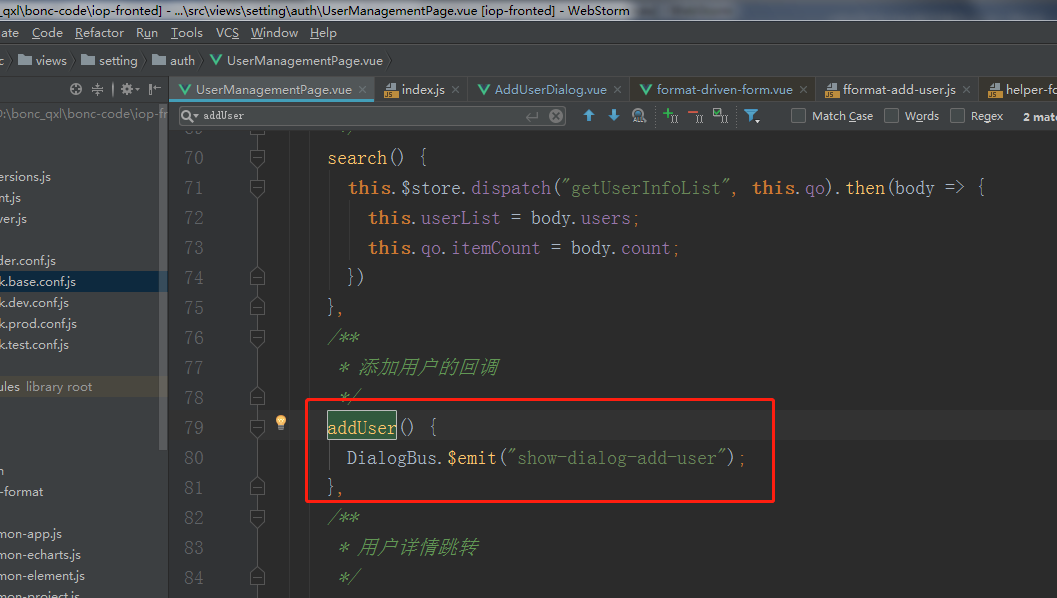
（1）以前：



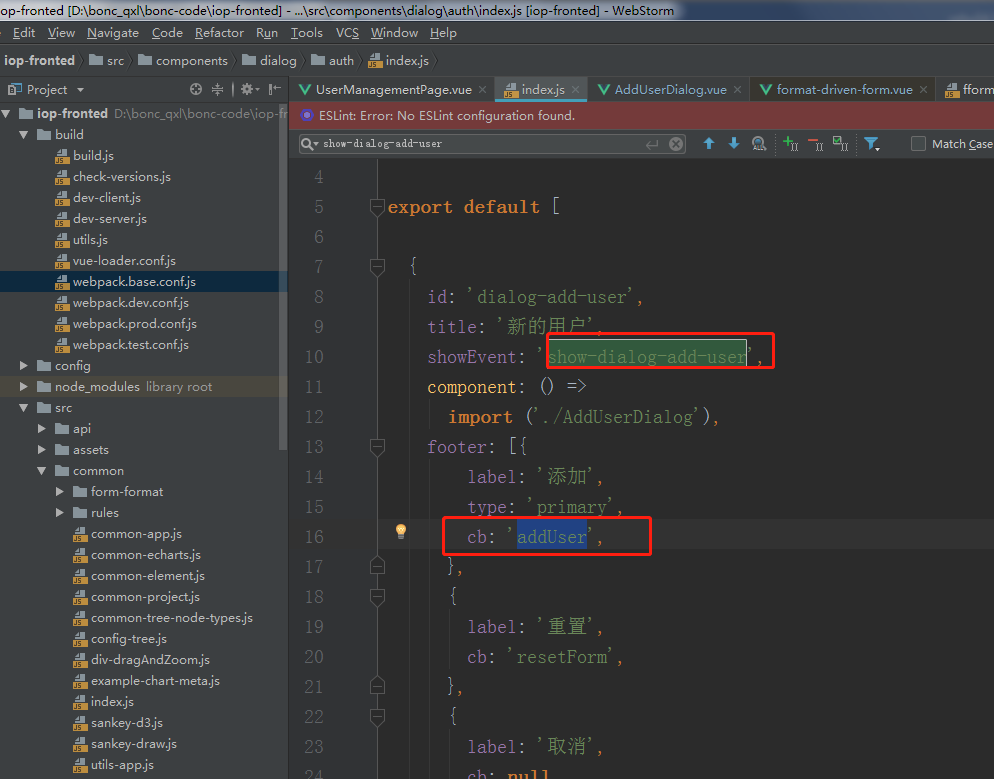


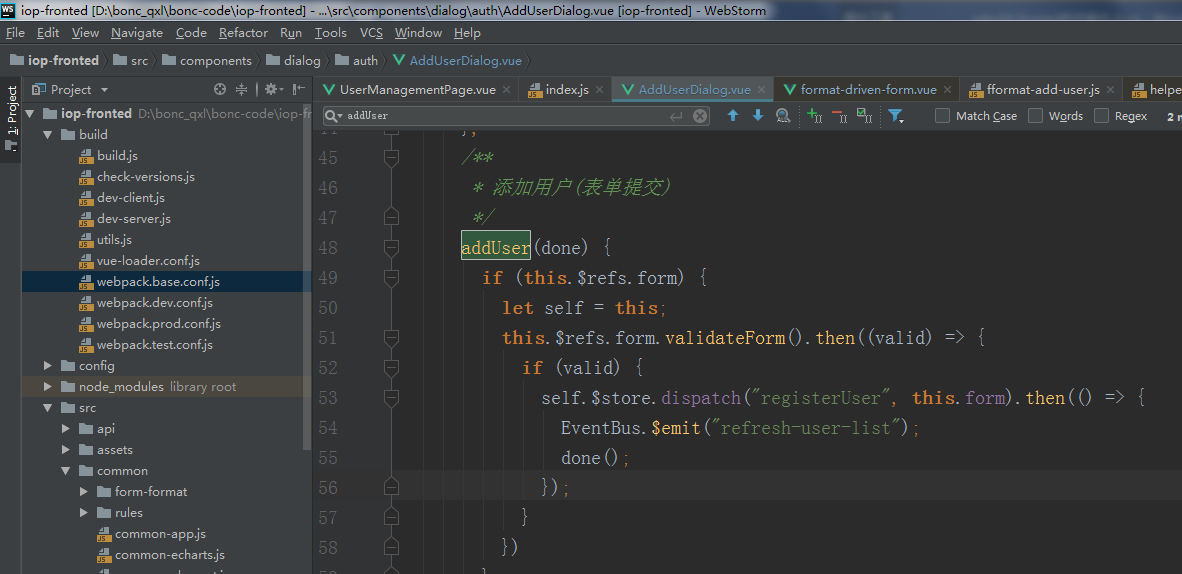
（2）重构后：





中间经过这个文件



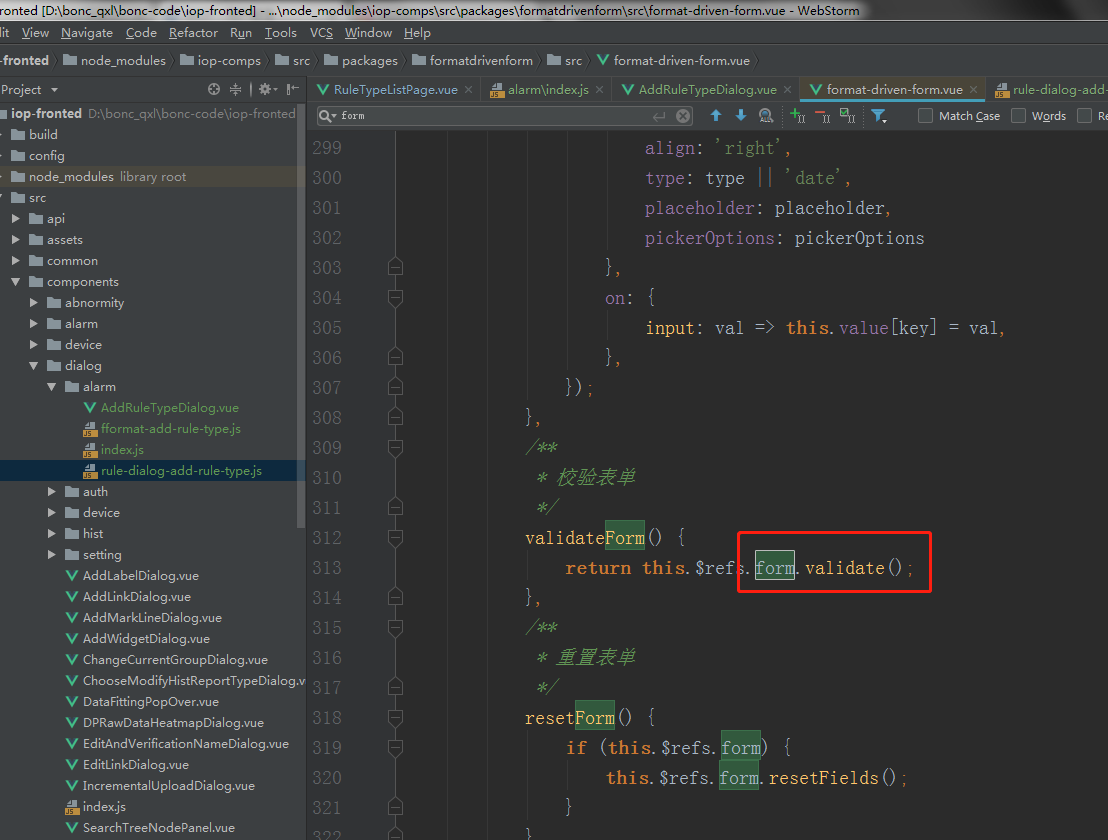


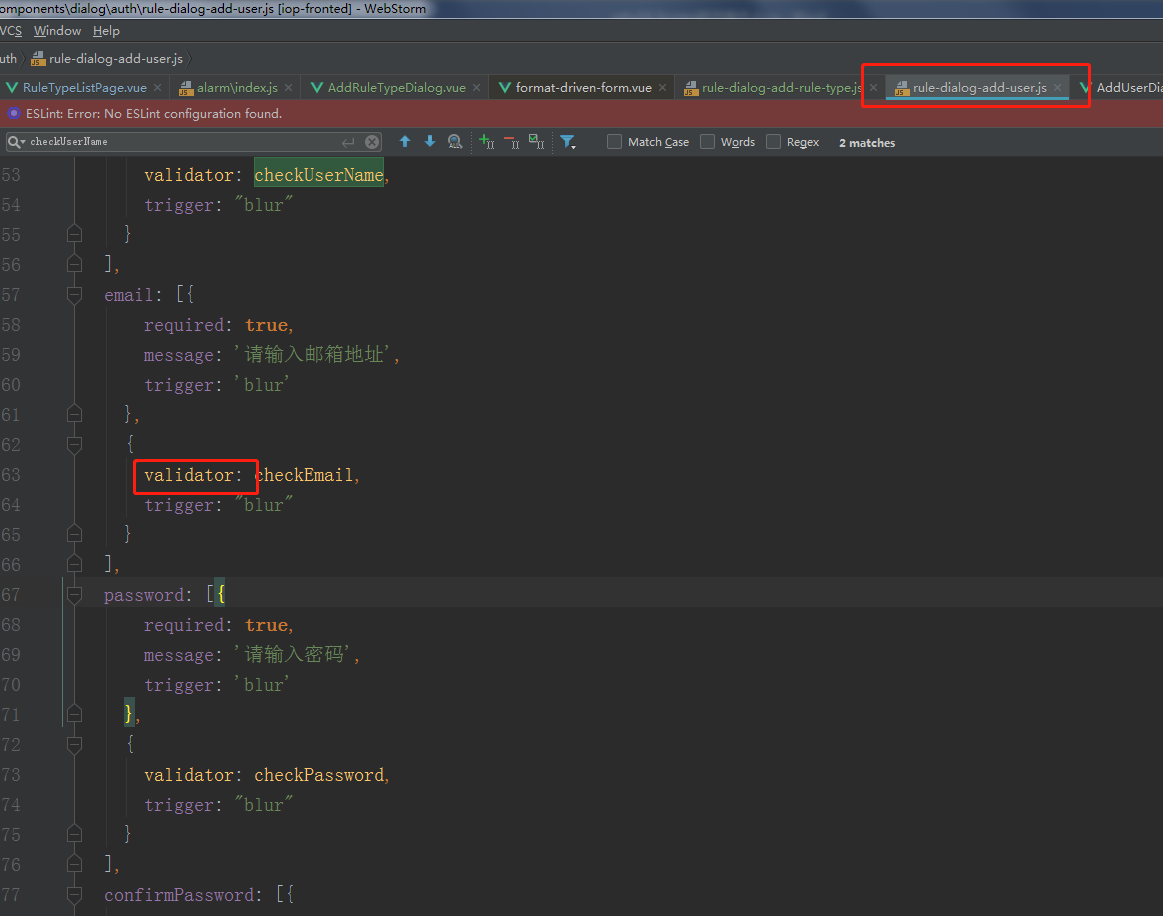
问题是这个addUser是在哪里调用了的？

5.



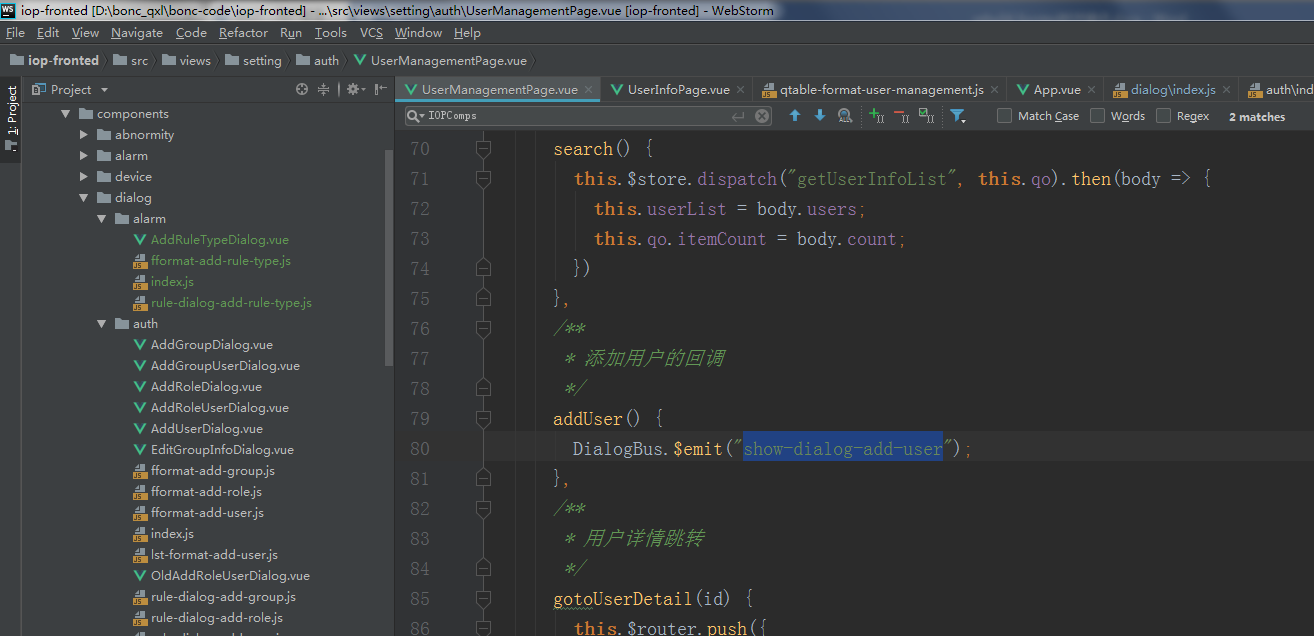
好好看看

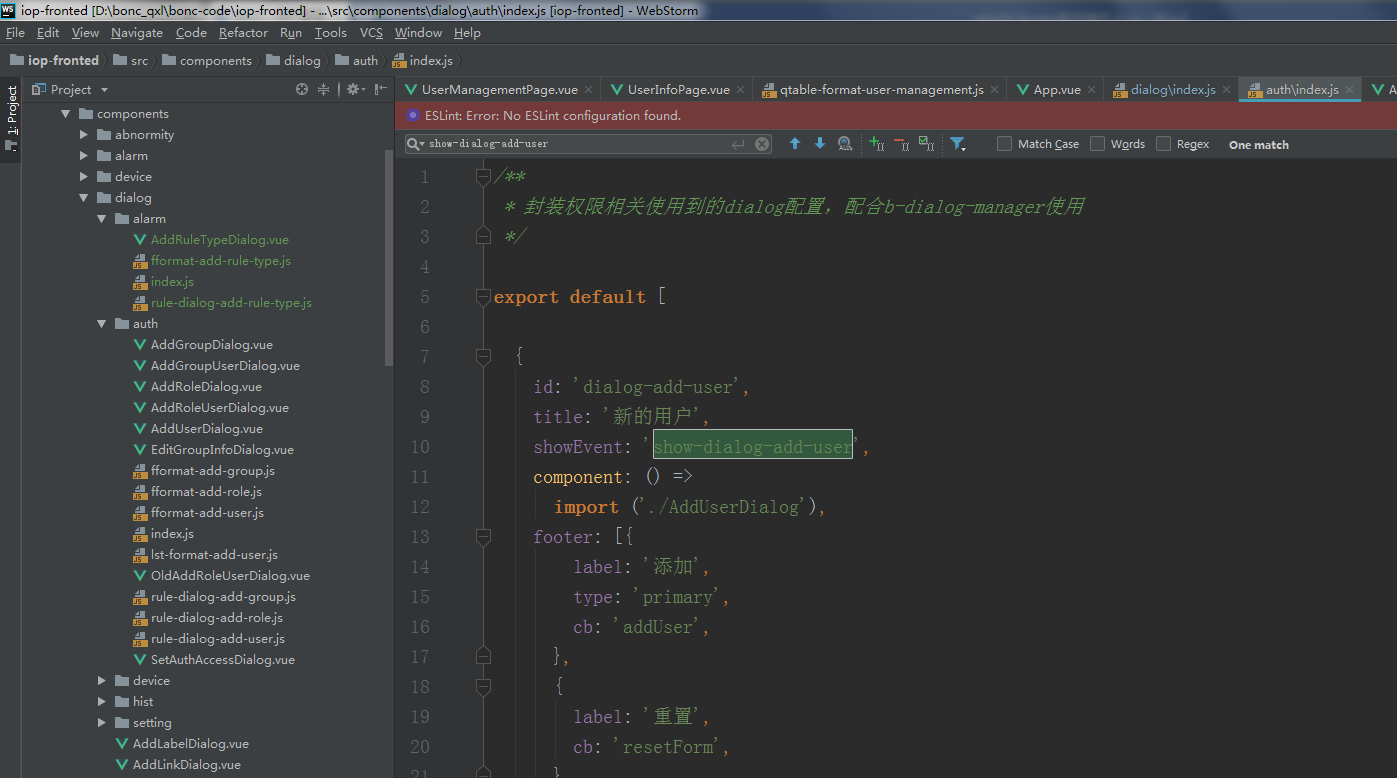


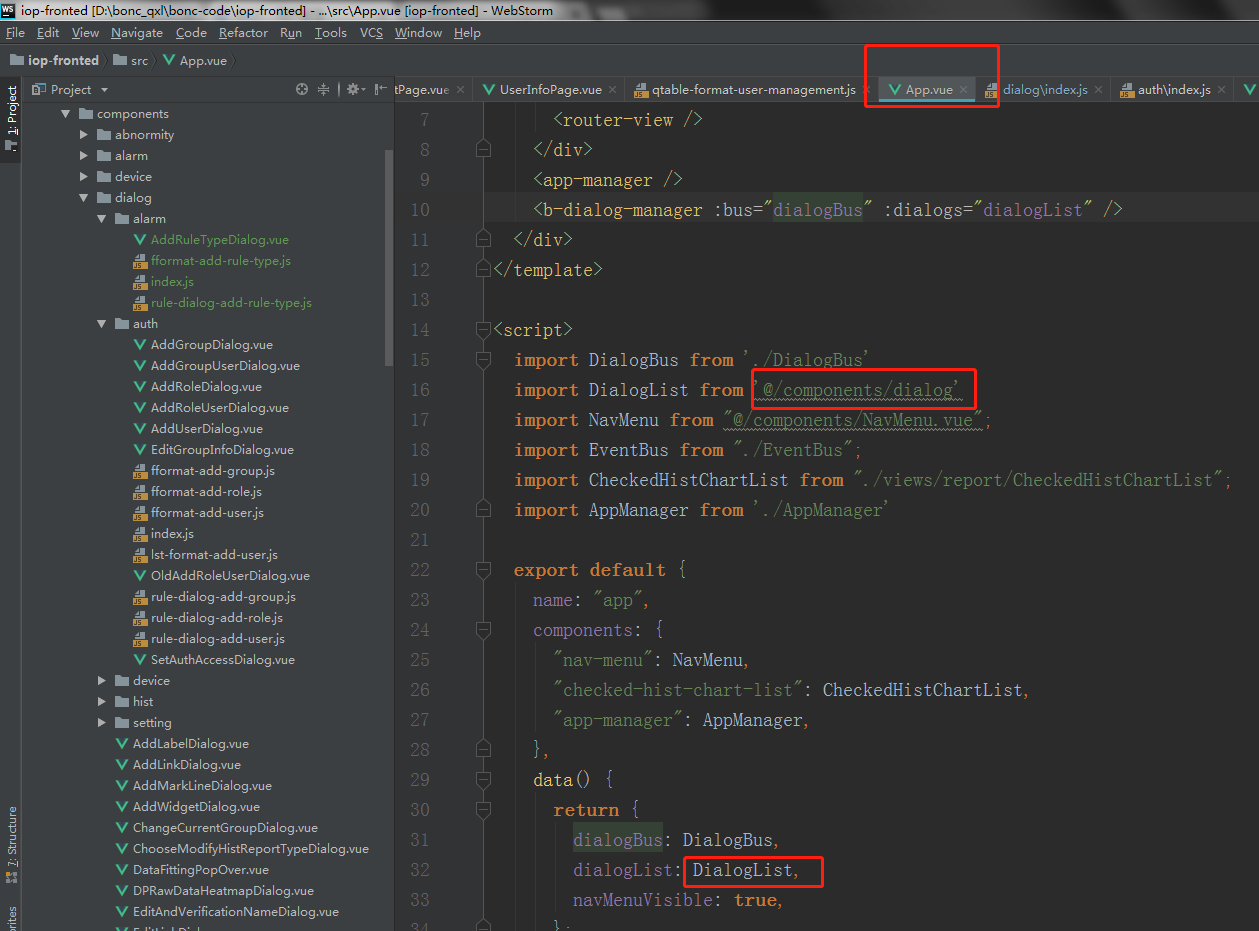


但是后者不在form里啊，是在rule里的

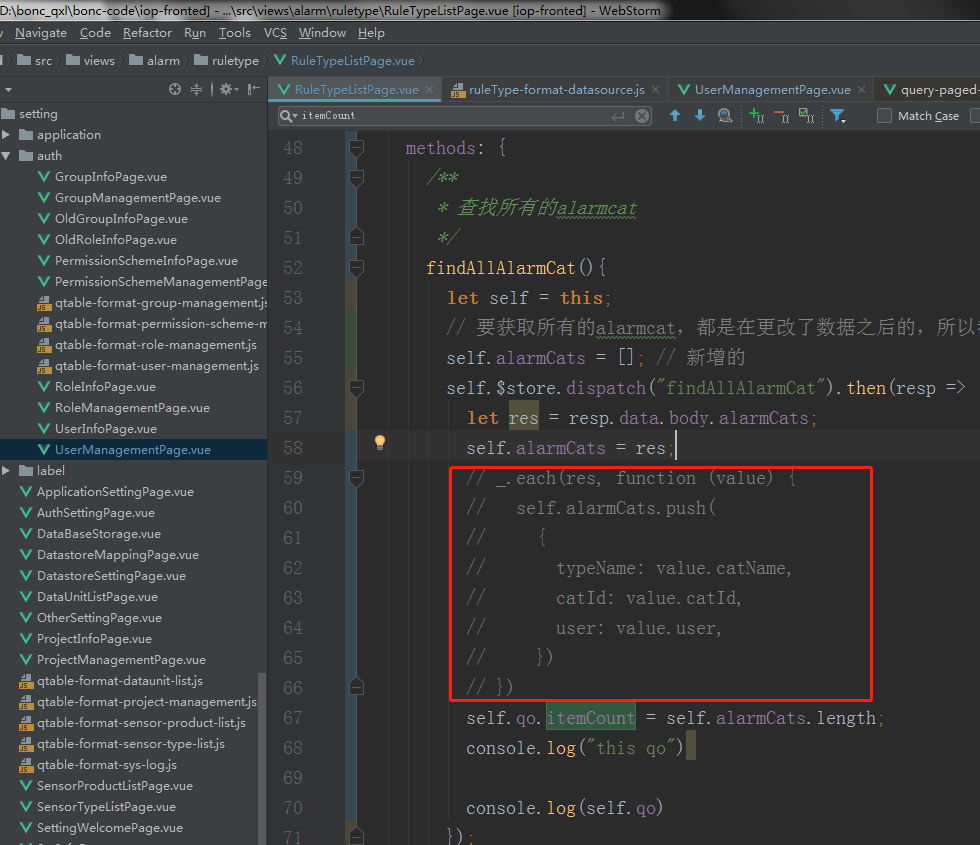
2.把这2个文件联系起来的是这个文件：



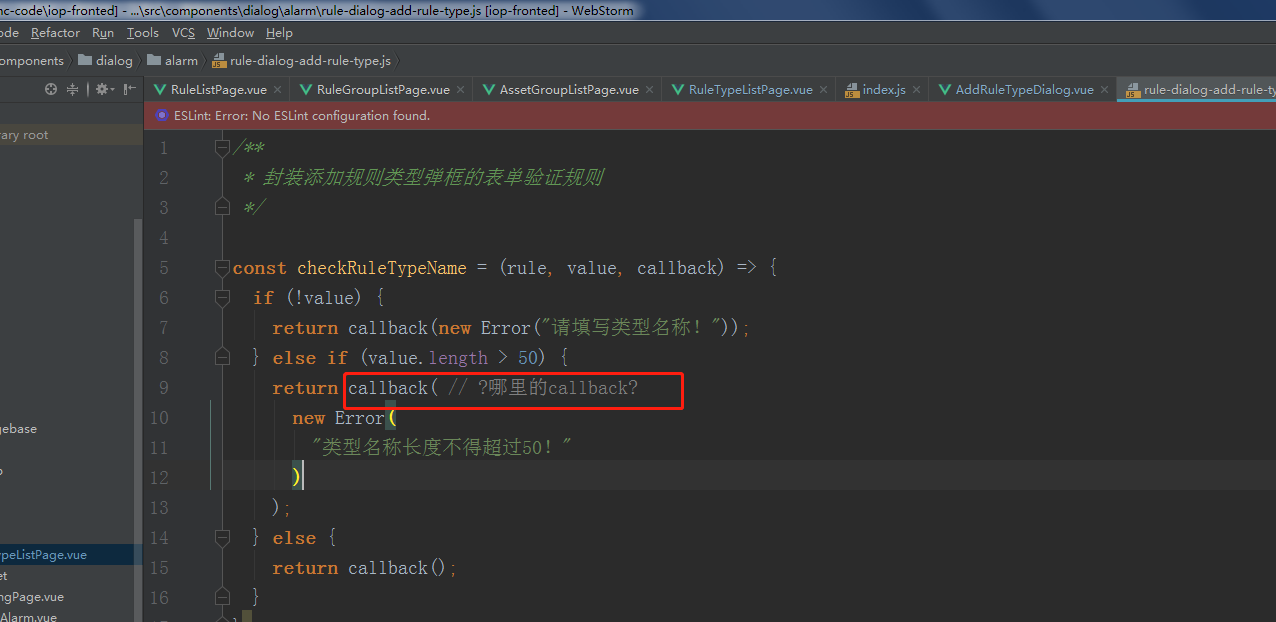




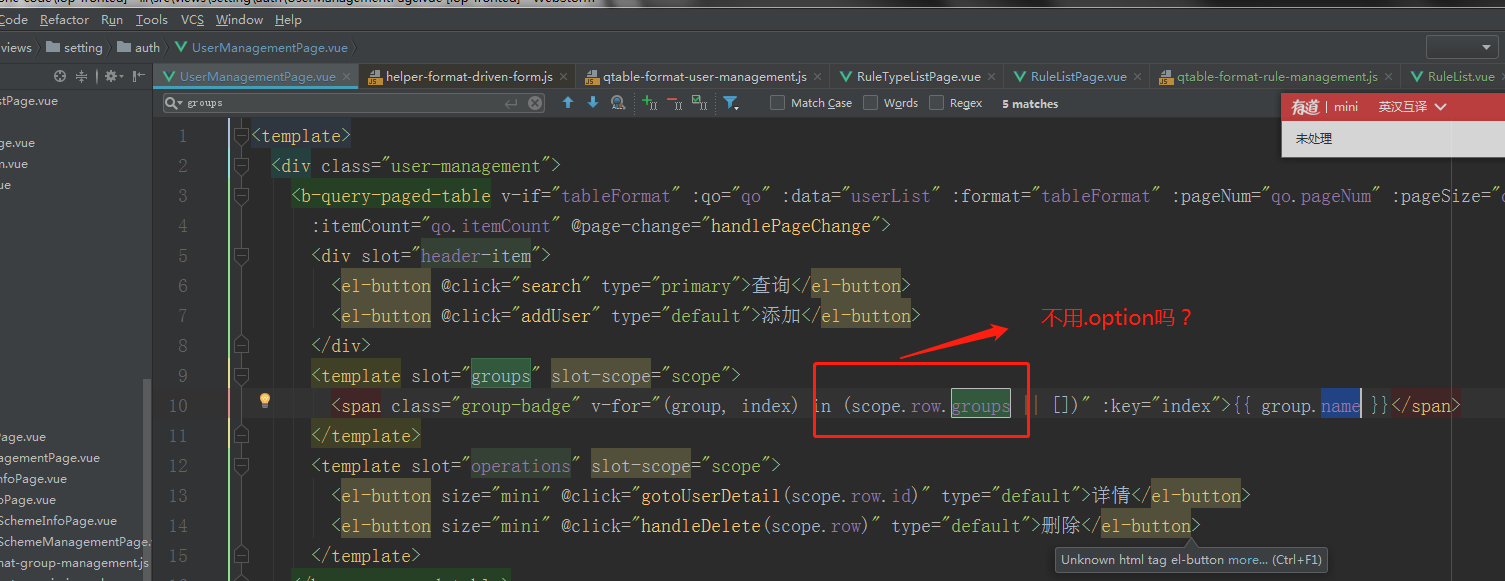
这样写是错的

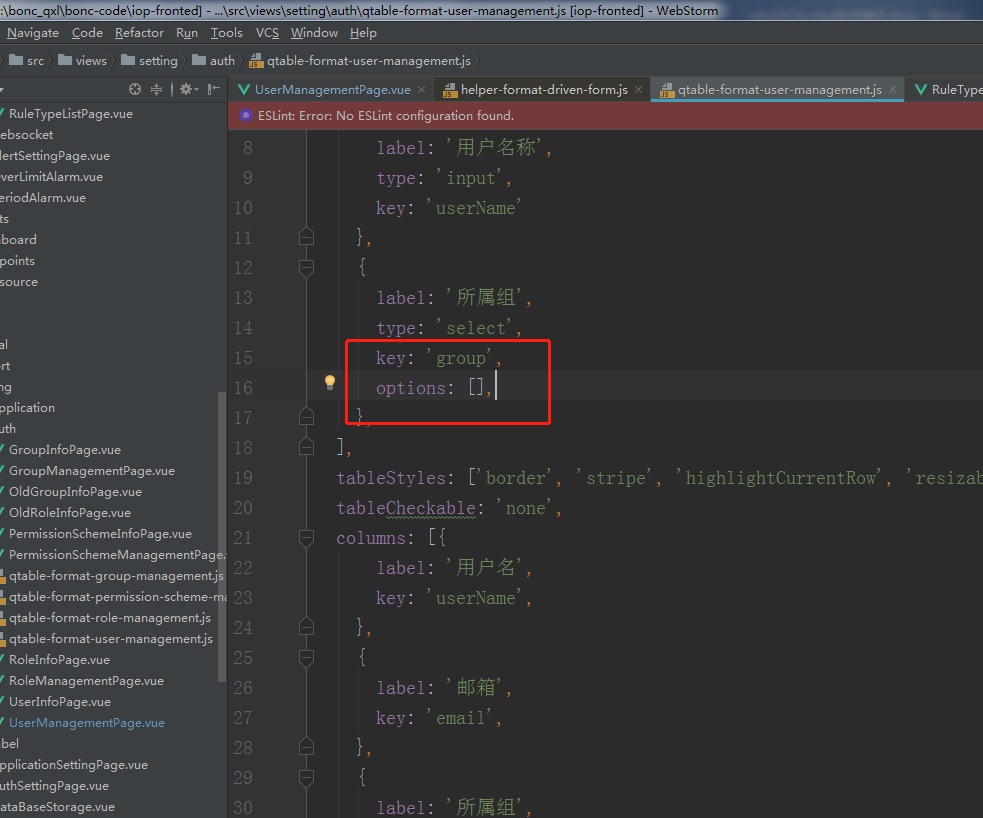


问题



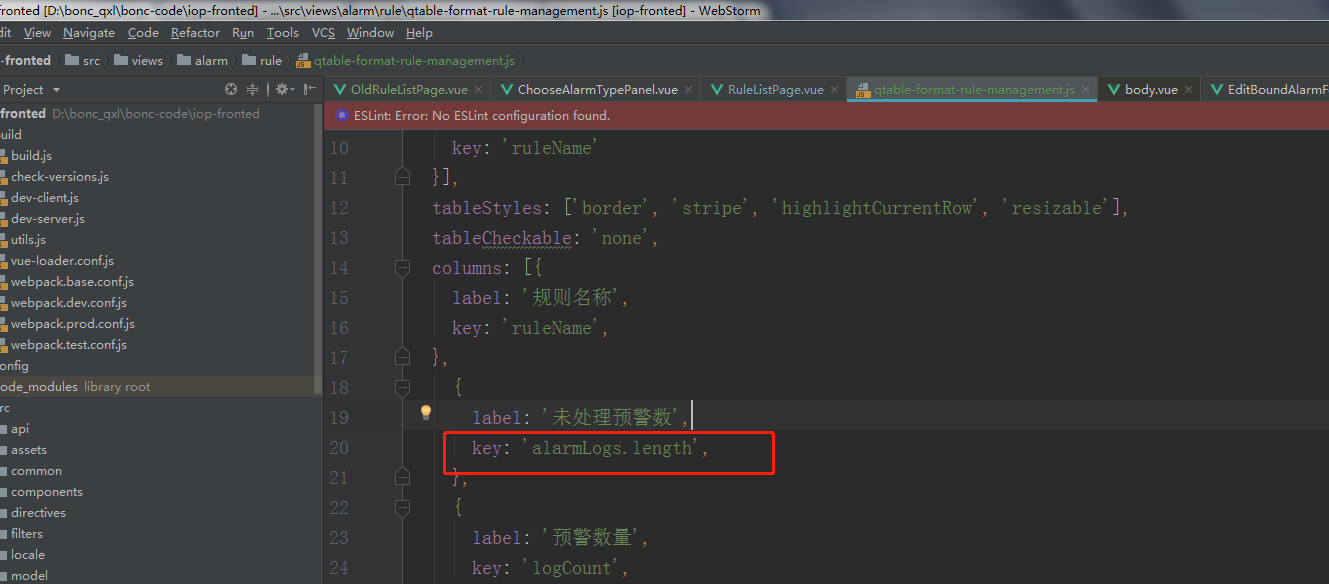
问题：



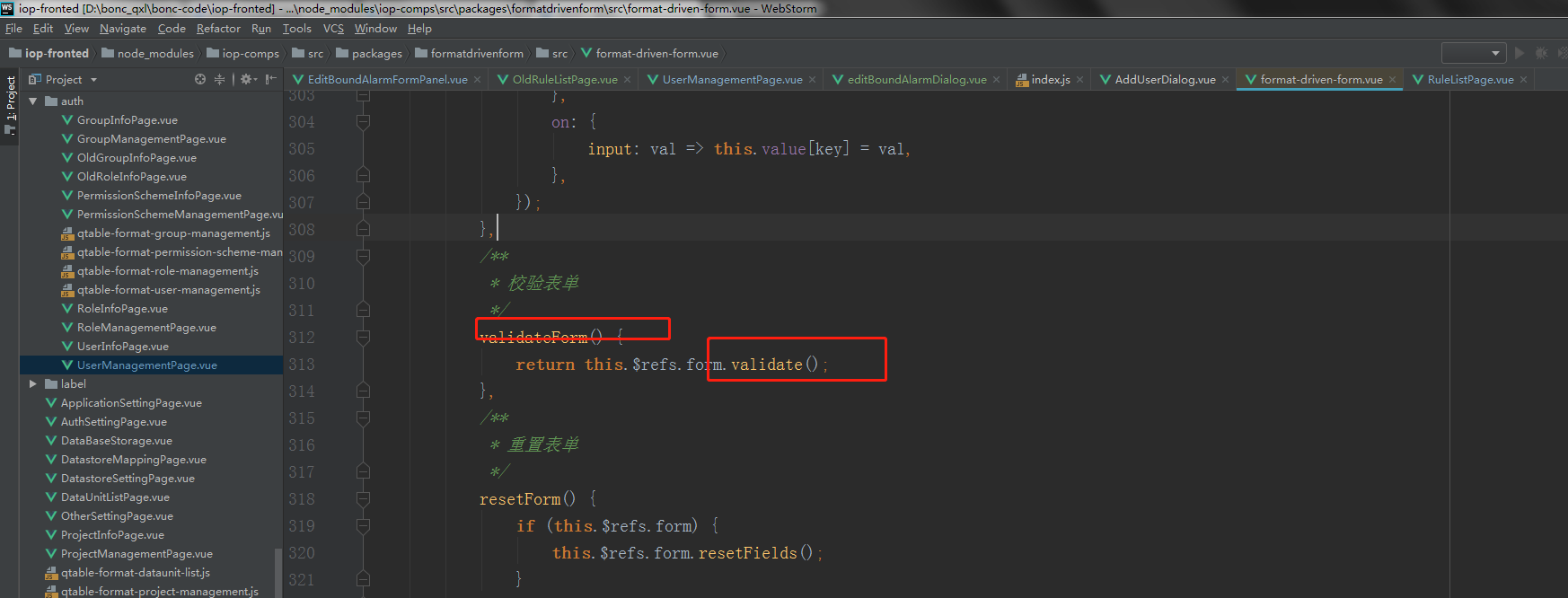


P132-607

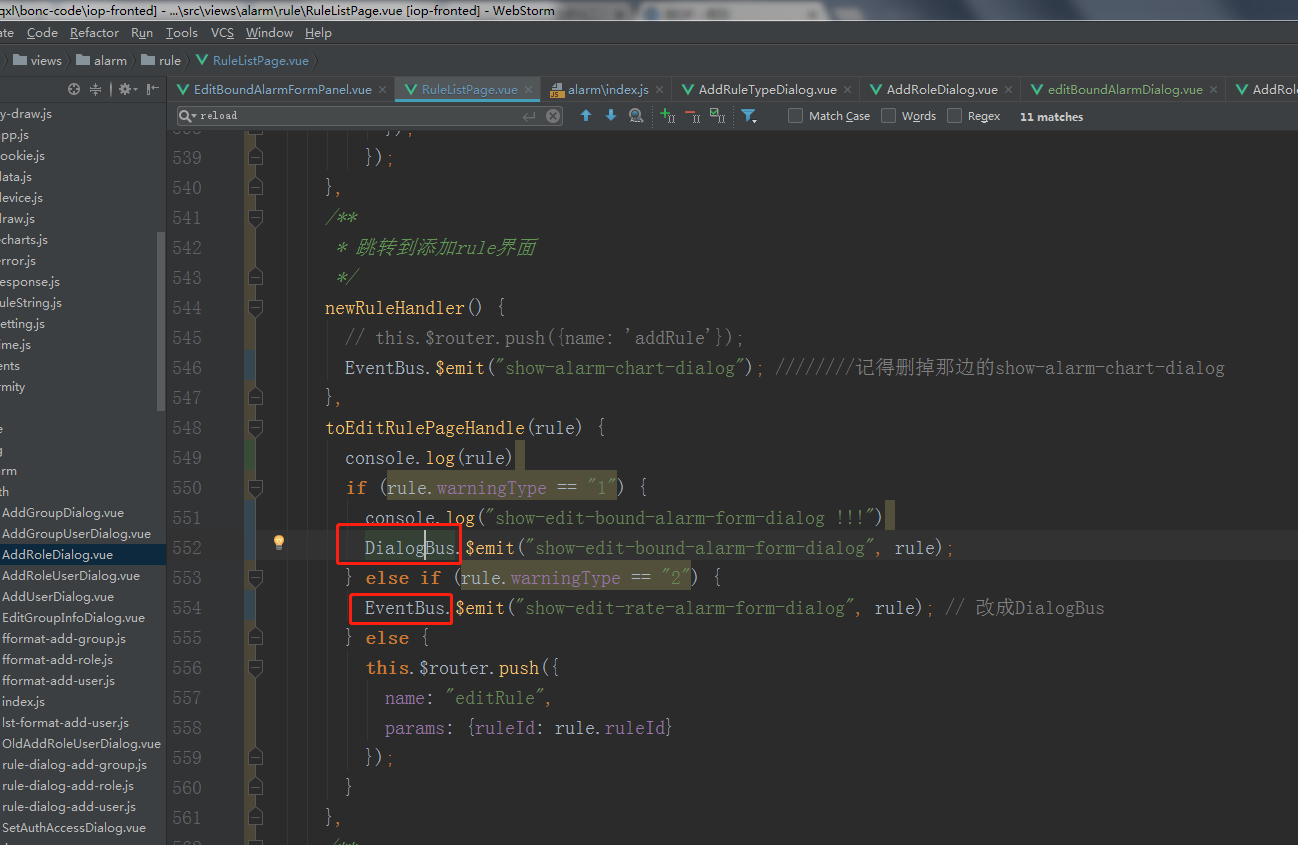
注意写法，好好理解



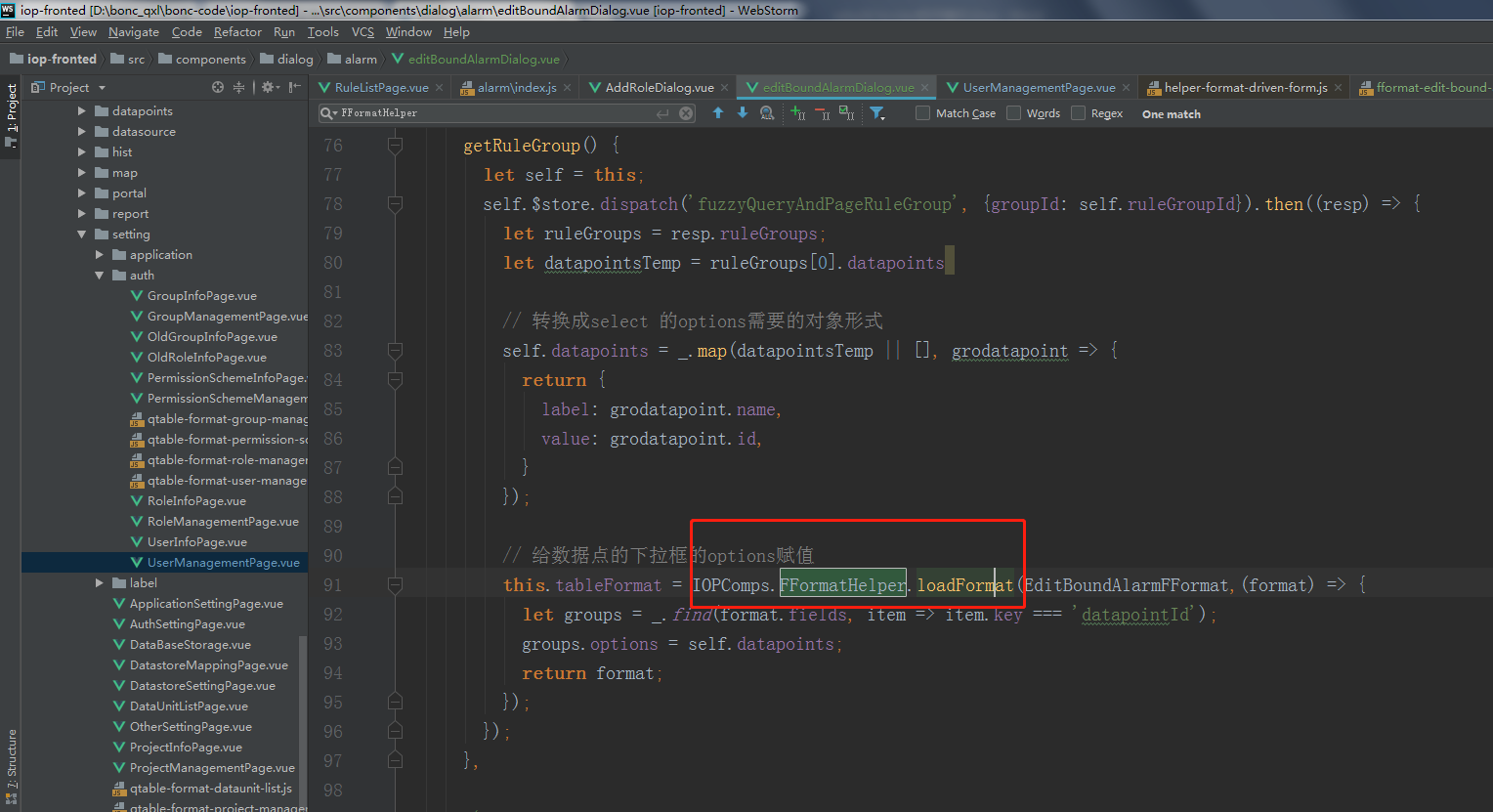
这里要搞清楚



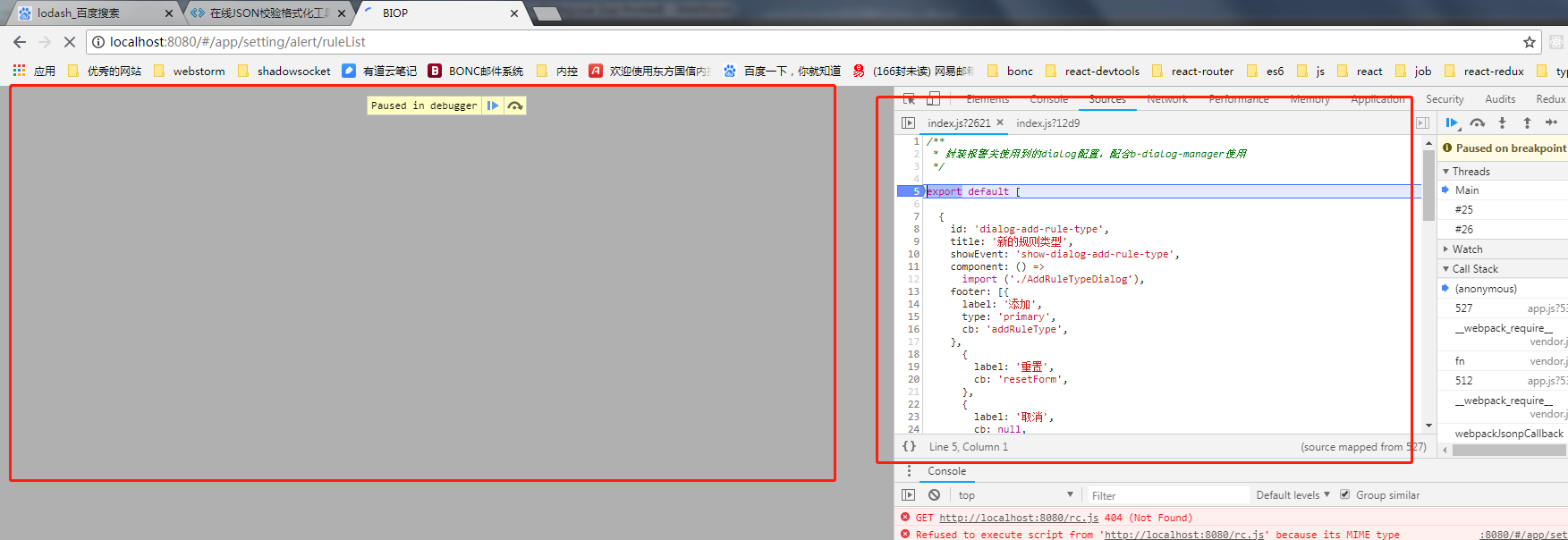
改的时候记得改成dialogbus,原来是eventbus



看看这个里面怎么写的：



对dialog的加载是在页面显示之前就加载了的，不是用到才加载，因为是全局注册的，多理解理解里面的精髓

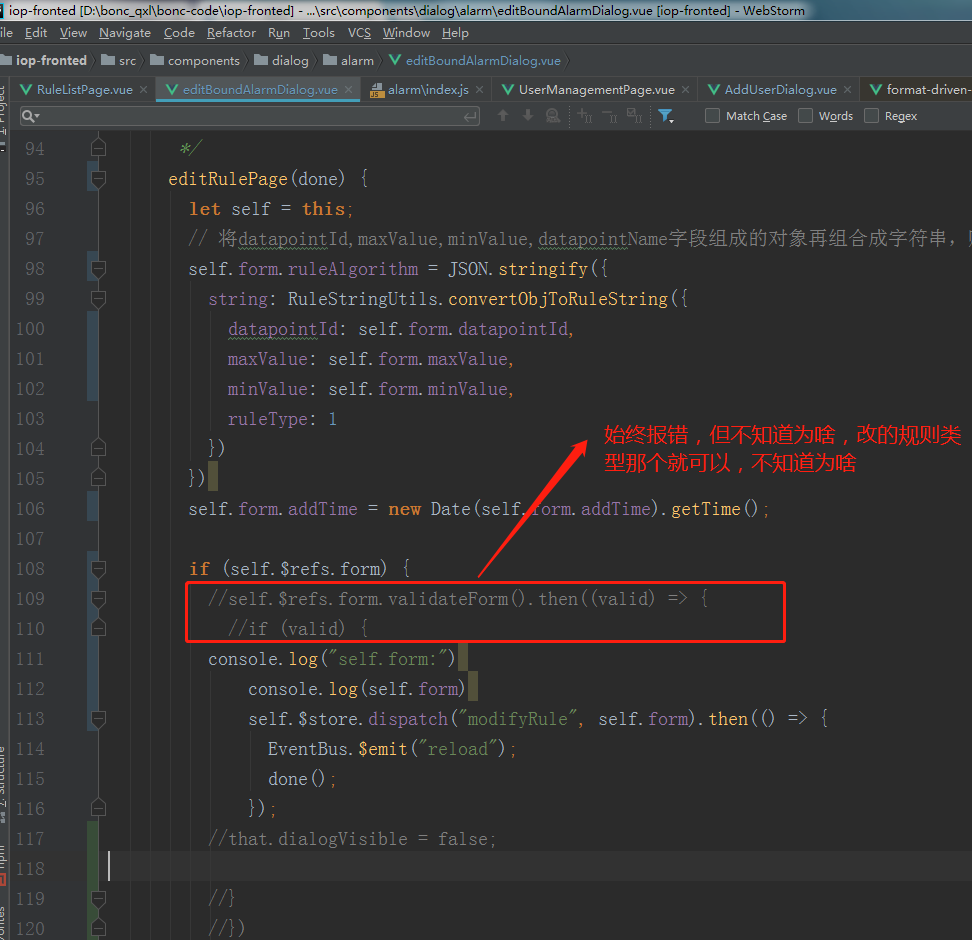


参考：



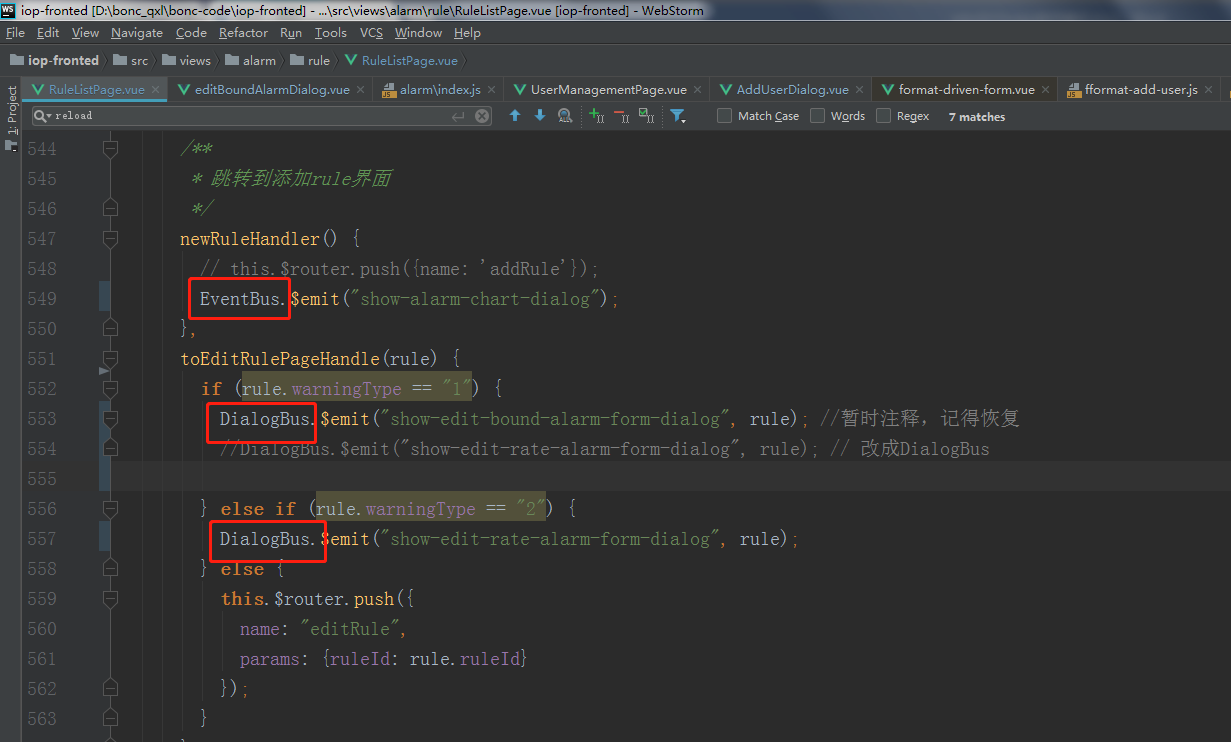


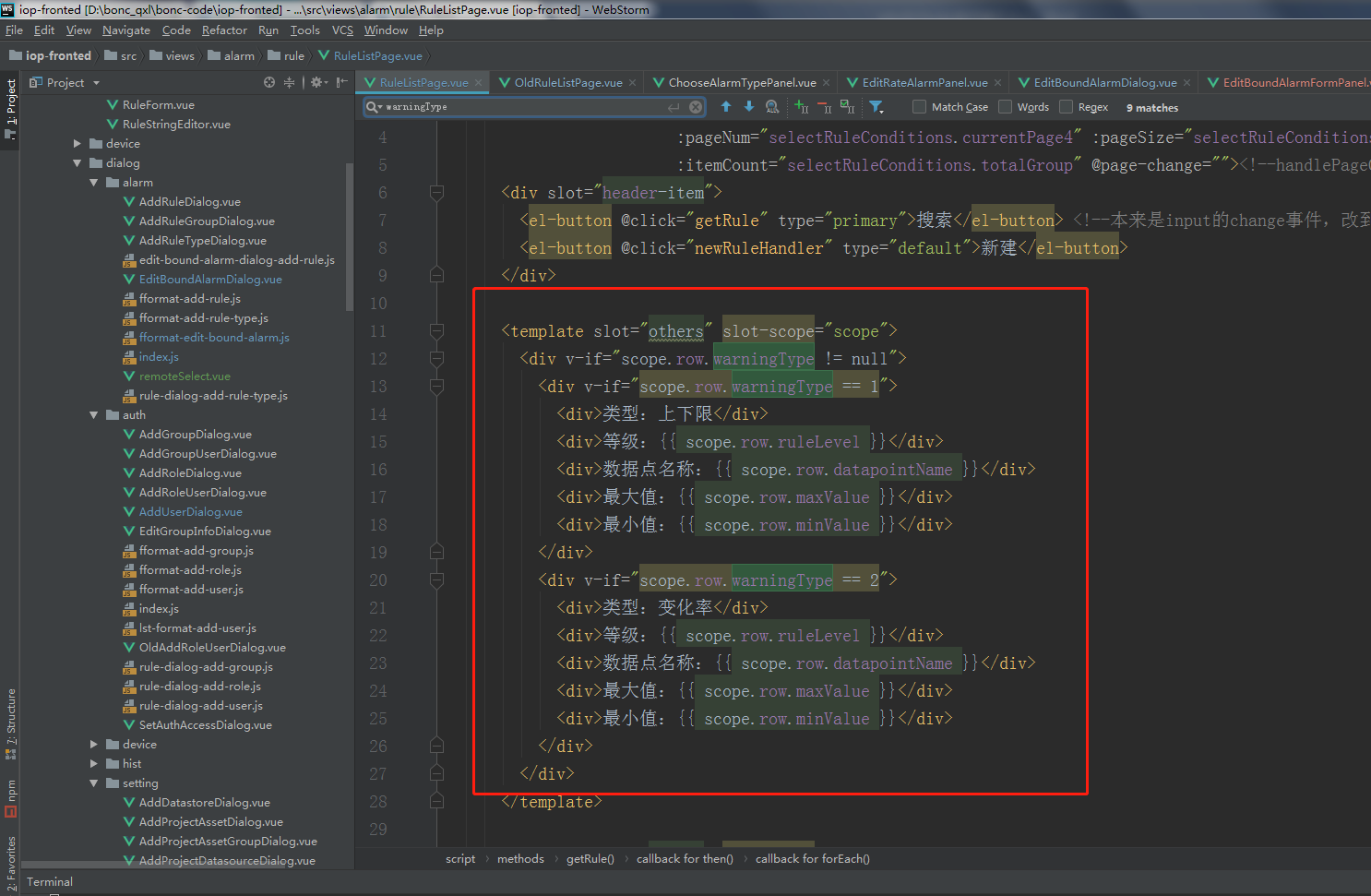
1. 有个问题：

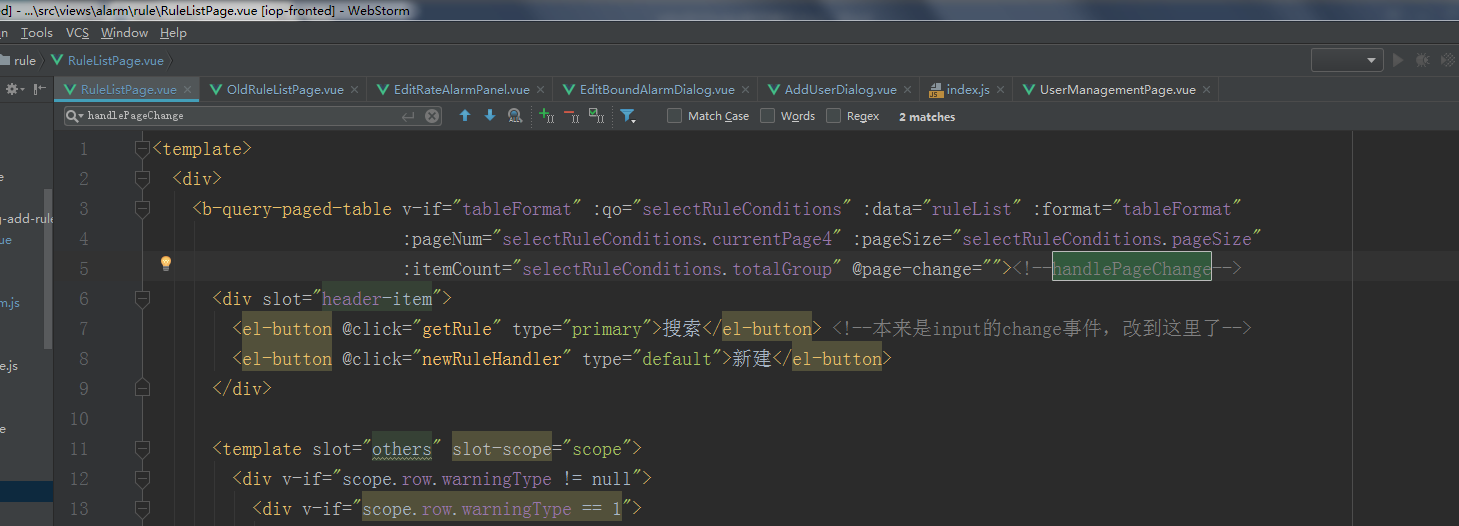


1. 有空看看这些

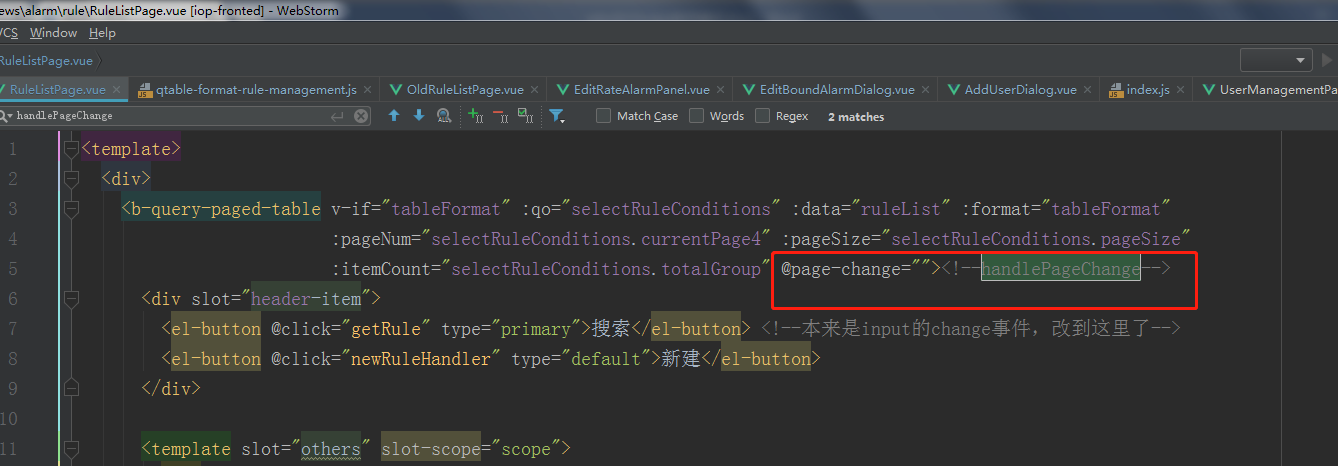




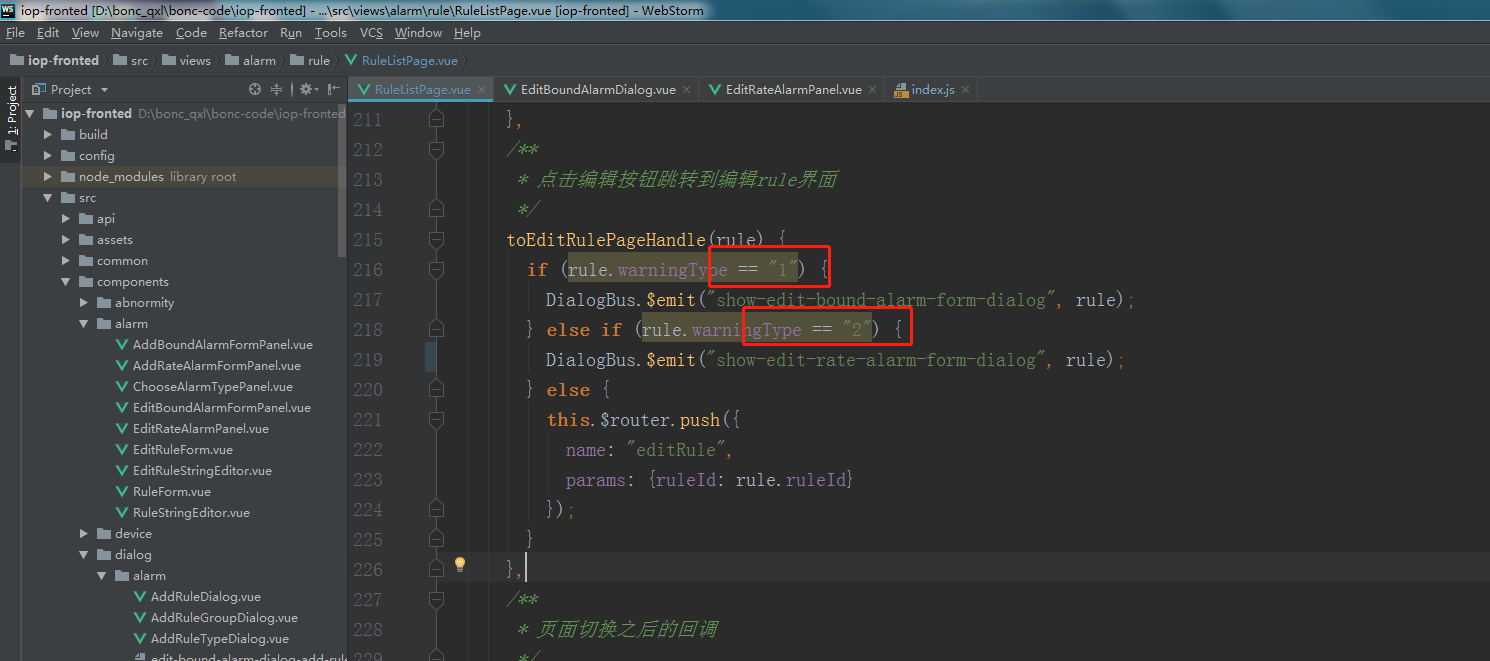




有空改一下这里分页，现在用的是原来的方法



这里若改成===，则会判断为false,不知道为啥



记录：

