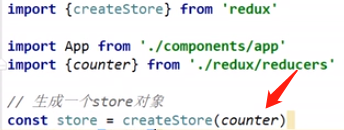


reducer的参数是state和action

根据不同的action处理不同的逻辑



用reducer创建store对象

C:\Users\林\AppData\Local\Temp\1568907720(1).png

export default时才能这么引入

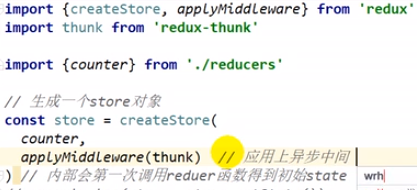
C:\Users\林\AppData\Local\Temp\1568907748(1).png

我现在以这种方式暴露

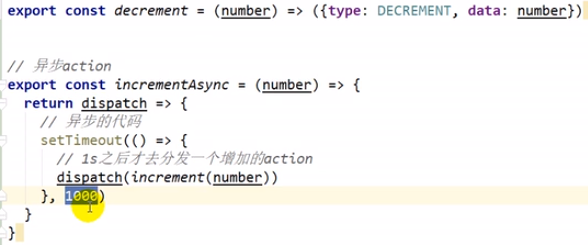
C:\Users\林\AppData\Local\Temp\1568907782(1).png

就必须这么引入或者{}结构引入

<https://www.bilibili.com/video/av51174155/?p=41>



thunk是redux可以进行异步处理

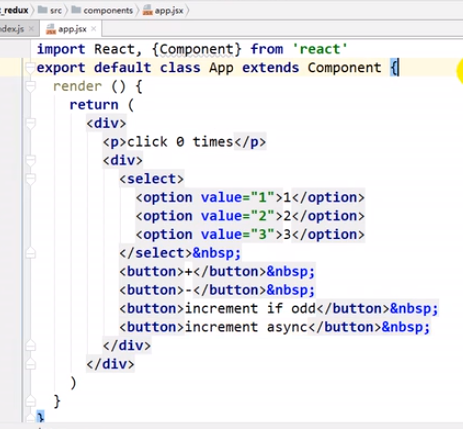


同步返回的是对象

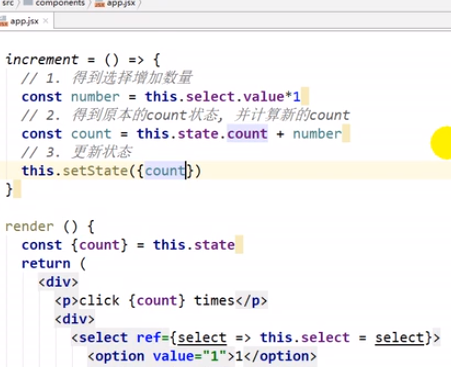
异步返回的是函数

redux调试工具

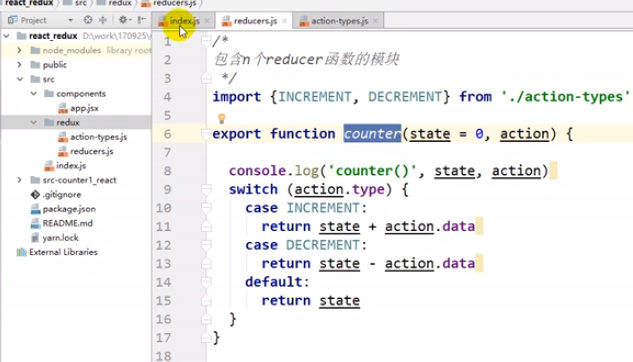
开始



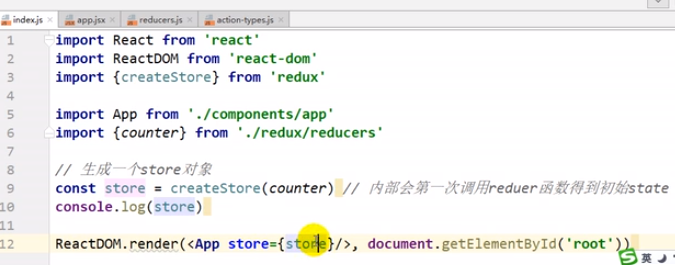
字符串转数字：”1”\*1





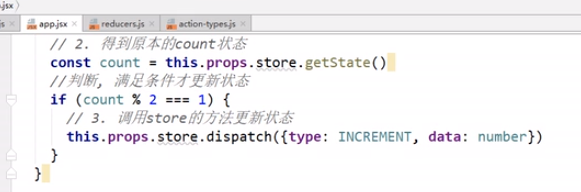


reducer包含state和action，处理不同逻辑



用reducer创建store对象

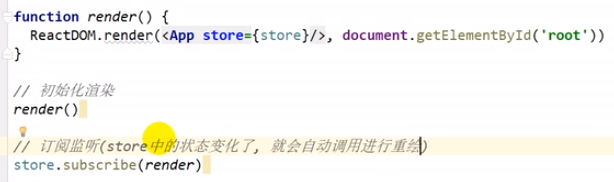
将store传给app全局应用组件

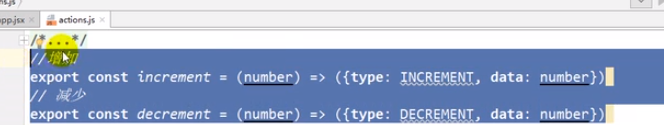


store放props中



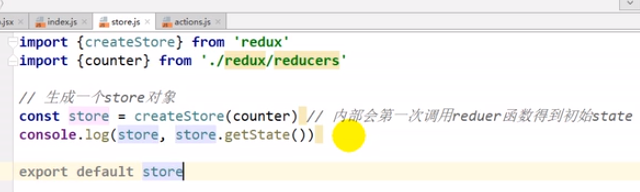
subscribe重新渲染组件(还不是react-redux)

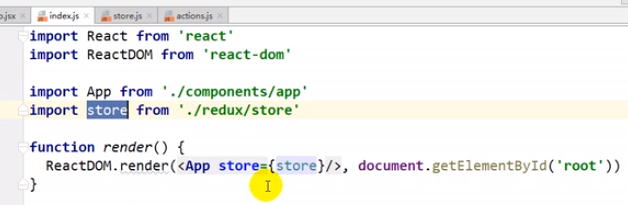




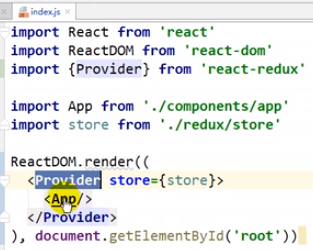
export default 的情况

这种和大括号的形式

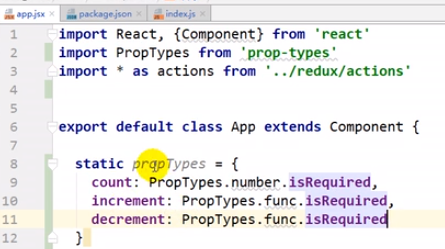




react-redux开始

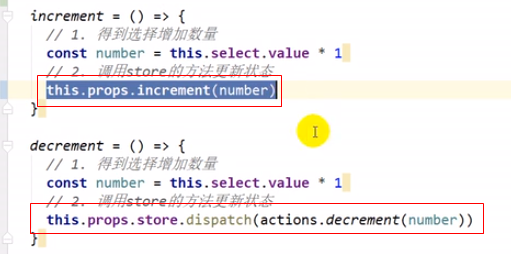


状态交给Provider管理,代码就可以少写，解放啦~~~

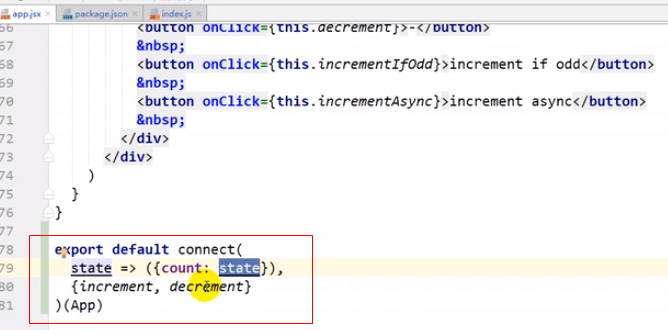


需要npm install

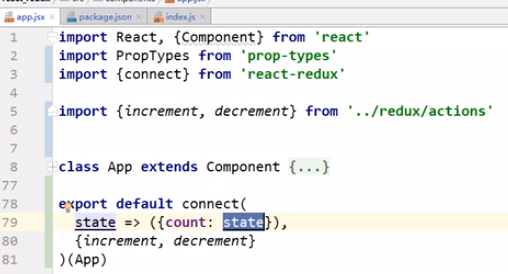
propTypes通过connect来连接（传递，赋予）



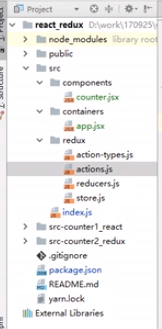
书写方式简单了

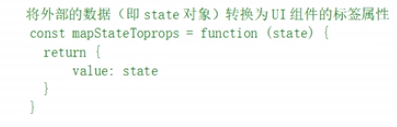


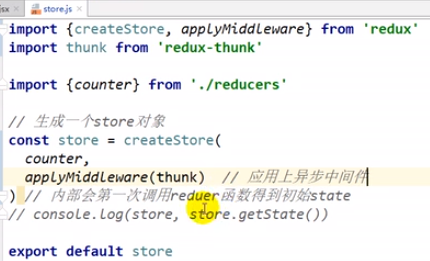
将store传递给App



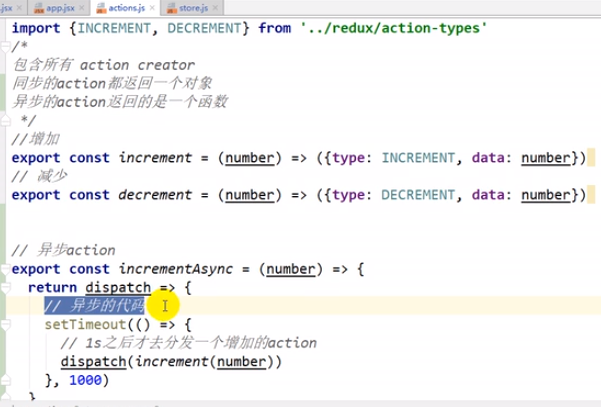
就是redux管理的state







thunk使redux可以异步处理



异步代码

react类组件的命名：不能使用驼峰，只能使用首字母大写