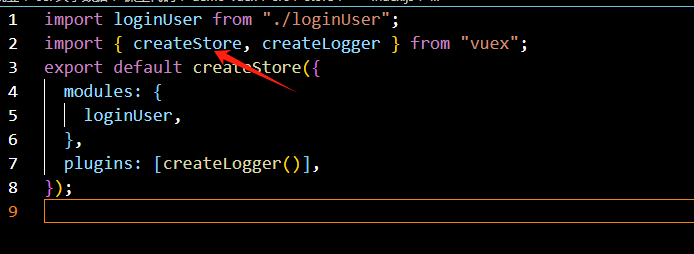
Vue3的共享数据

所谓的共享数据就是响应式数据完全可以不去依赖vuex

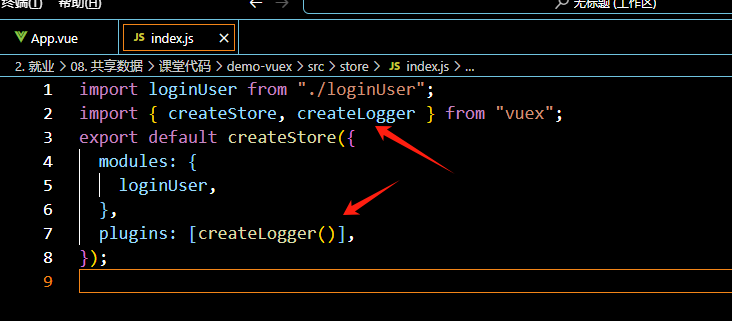
Vuex方案

首先想要在vue3中使用VueX，那么就需要安装Vuex@next 最新版本 npm i vuex@next

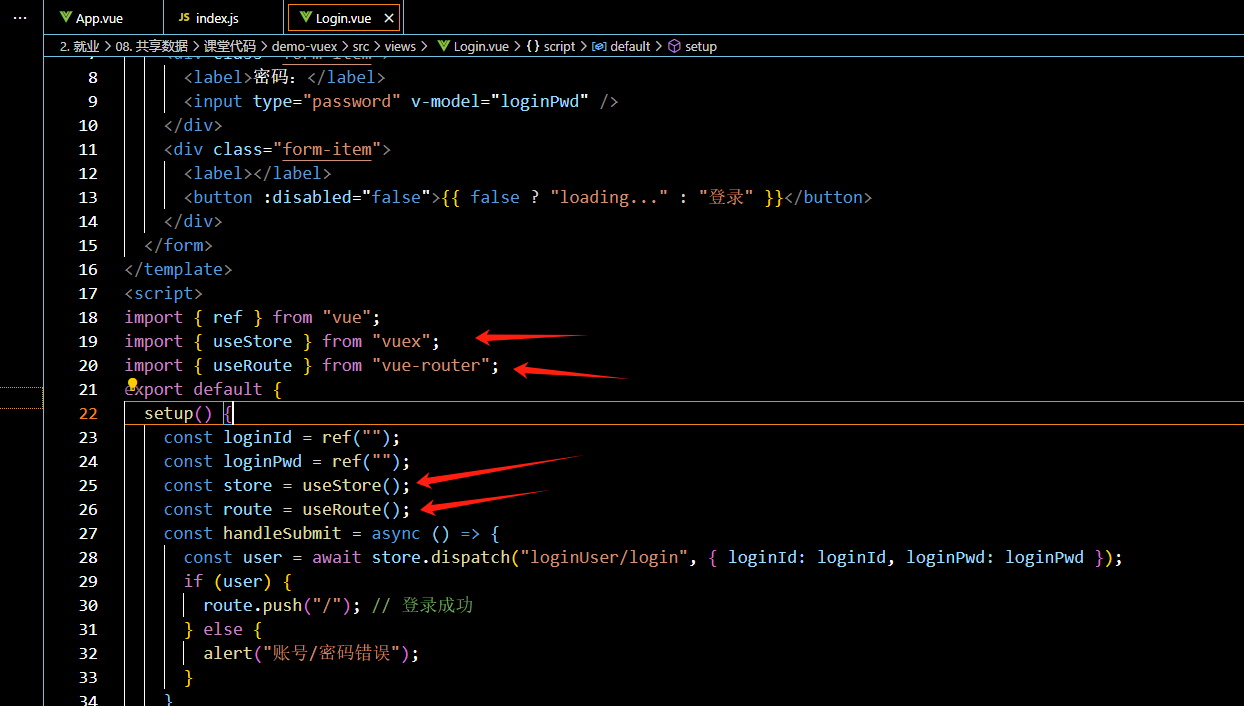
Vue3的vuex创建仓库使用createStore创建的



在VUEX中可以配置createLogger 它是用来查看日志的就是你vuex发生变化了它会在控制台输出



由于vue3没有this 引入 vuex和router 要单独用useStore useRouter 去模块中取 就是比vue2多加了个use



Global state 方法

由于vue3的响应式系统本身可以脱离组件而存在，因此可以充分利用这一点，轻松制造多个全局响应式数据。

比如:

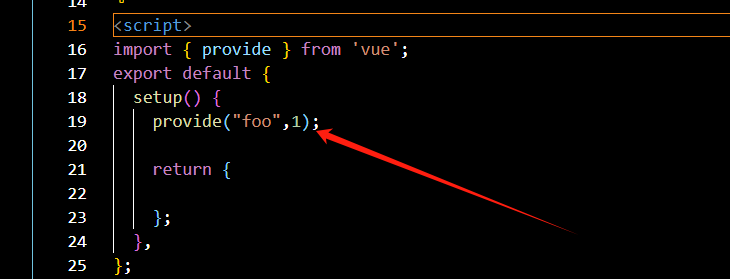


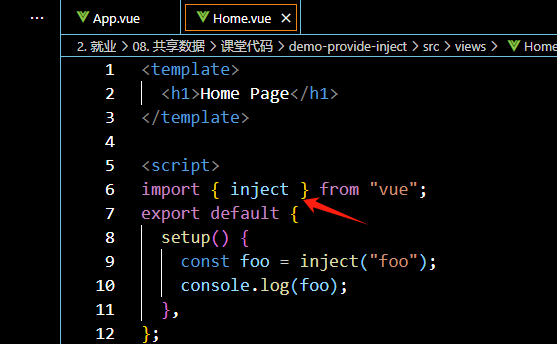
Provide&inject方法(注入和使用) 类似vue2的 父子组件传参

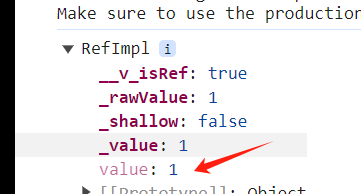
Provide给父组件用，inject 给子组件用

其实vue3兼容option配置的形式，但是vue3没有this 所以建议不要用，vue3有composition api 它里面增加了provide和inject 可以在setup注入和使用数据

比如:







如果说我要给整个vue应用提供数据呢？比如vue2 的event bus ，vue3没有这种api 但是可以做到同样效果就是provide 和inject,给全局提供数据，然后用inject注入数据在组件中引入使用。

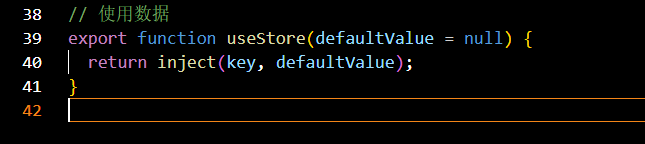
这里vue3给我们提供了一个方法可以在createAPP()创建应用的时候去使用 .provide(“key”,ref(1))

比如:

我这里给你导出一个对象 供对应的vue应用使用，如果是不同的vue应用那么你给我传不同的应用即可

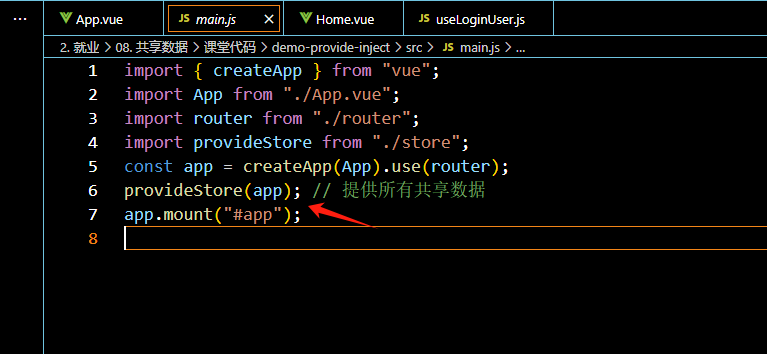


我上面的全局数据给你了之后 我提供一个方法来使用它 直接调用这个方法即可



进一步的去封装 全局数据的方法 提供统一的数据接口





对比:

1. 如果我只是想做组件的数据共享那么哪一种都可以

2. 如果我脱离组件存在 那么vuex和global state 可以脱离组件存在，provid e 和 inject只能在组件中去使用，它不能脱离应用存在，如果脱离比如main.js 那么当他运行的时候组件还没有加载完成。

3.调试工具。vuex支持 其他两种自行决定。

4.状态树 vuex支持 其他自行决定

5.量级: vuex 重 其他两种 轻量级