**Redux**

**1. redux 基本介绍:**

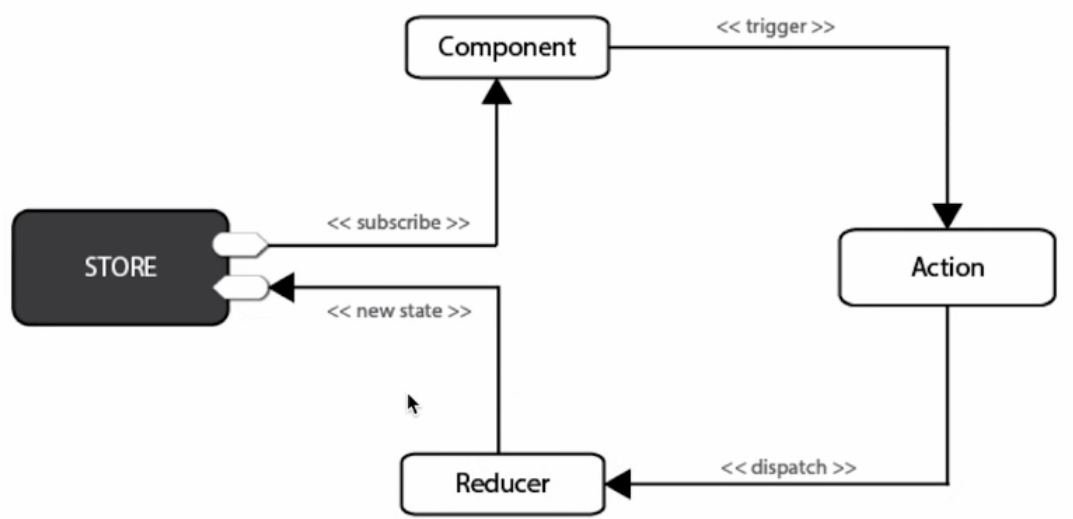
**- react中为单向数据流: 从父组件流向子组件,且兄弟元素中无法共享数据**

**- State :只读状态 不可修改**

**- Reducer:基于函数用于对State的业务处理**

**-Action:用于改变state,普通对象,用于描述事件行为**

**2. Redux 工作流程:**

****

**3. Redux安装:**

**- yarn add redux --save**

**- yarn add react-redux –save**

**Chrome调试插件 redux-devtools-extension**

**4. Redux集成:**

**-创建Action模块**

**-创建Reducer模块**

**-创建Store模块**

**-通过Connect方法讲React组件和Redux组件连接起来**

**-添加Provider作为项目的根组件,用于数据的存储**

**5. 基本概念和API**

**--5.1 Store**

**-Store:保存数据的地方,可以将他看成是一个容器,且整个应用只允许有一个Store!**

**-createStore: redux提供的这个函数 , 用于生成Store**

**--5.2 State**

**-Store对象所包含的所有数据! 如果想得到某个时点的数据，就要对 Store 生成快照。这种时点的数据集合，就叫做 State。当前时刻的 State，可以通过store.getState()拿到。**

**--5.3 Action**

**-State的变化会导致View的变化 , 但是用户接触不到state,只能接触到view , 所以,state的变化必须是view导致的 . Action就是view发出的通知,表示state应要发生变化了!**

**-Action是一个对象,其中type属性是必须的 , 表示Action的名称,其他属性可自由设置;**

**--5.4 store-dispatch**

**-store-dispatch是view发出Action的唯一方法!**

**-- 5.5 reducer**

**- store收到Action后,必须给出一个新的state,这样view才会发生变化这种state的计算过程就称之为reducer;**

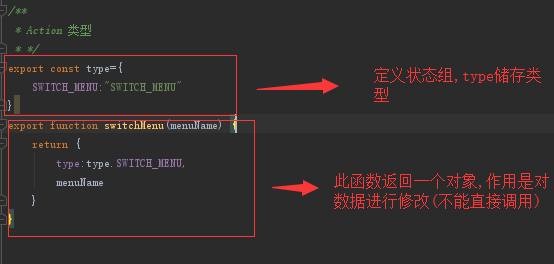
**-reducer是一个函数,接受Action和当前state作为参数,返回一个新的state**

**实例:**

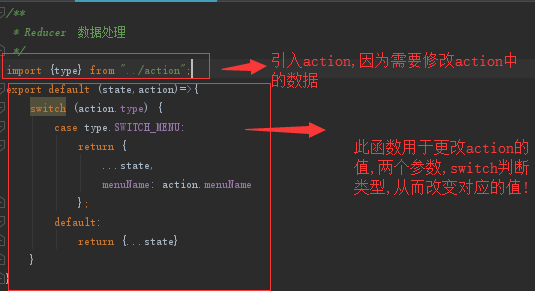
**步骤:**

**1. 创建action文件**

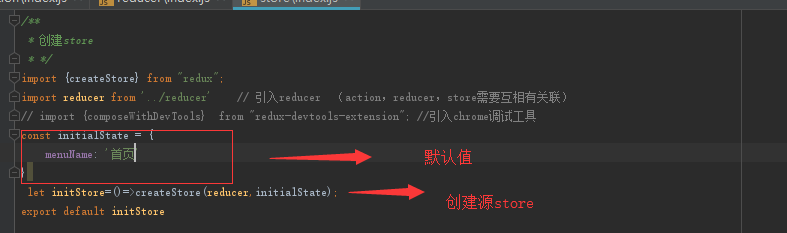
**此文件定义类型type和其他数据,type字段是必须的!**

****

**2. 创建reducer文件**

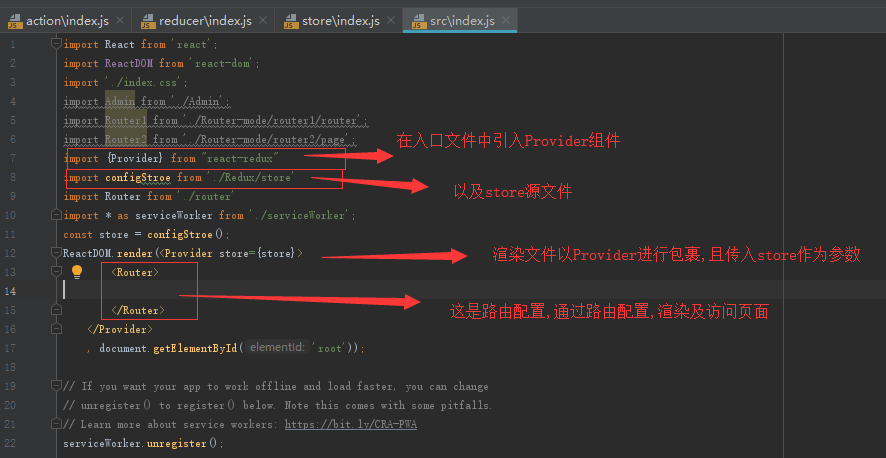
****

**3. 创建store文件**

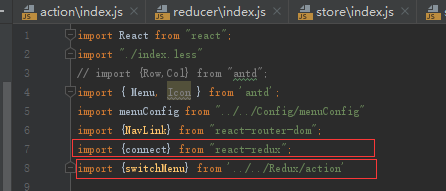
****

**4. 开始使用**

**在入口文件中需要引入对应的组件,如Provider ,此组件作为入口组件,并传入store作为参数,图解如下:**

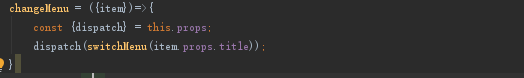
****

**5. 以左侧导航为例:**

****

**在nav中导入connect,来关联UI组件和redux数据,以及action中的所定义飞函数来设置数据**

**如在nav点击的情况下,需要改变菜单的名字,这里的type就是对应存储当钱菜单名飞类型,里面的值根据点击的位置的参数,来更改!如下:**

****

**这是menu菜单的点击事件,首先,需要使用dispatch,然后通过dispatch来执行menu更改函数,并且告知reducer,准备更改保存的状态数值,此时reducer同样会执行,会根据type将数据正确的写入对应类型中去!**

**6.在显示处:**

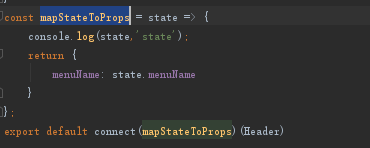
**因为需要使用到redux,所以此时需要引入connect进行关联!**

**此例是在header中显示面包屑导航的标题:**

** 在header组件中,将此处的值更改为:**

****

**需要定义一个mapStateToProps传入connect:**

****

**bingo ~~~**