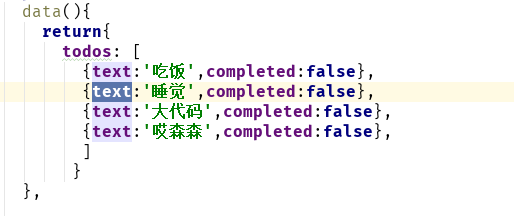
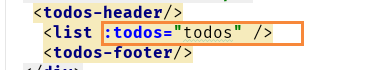
****

**首先要定义组件，拆分静态组件**

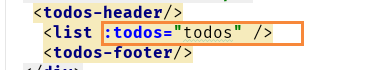
**在App中定义数据**

****

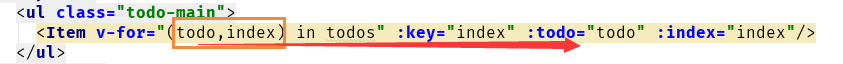
**在list标签中加上这个属性来传入列表中**

****

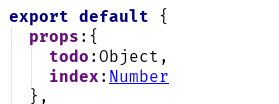
**在list列表中声明**

****

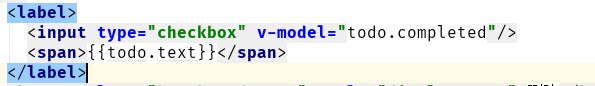
**在list列表的item中使用数据**

****

**再将todos里面的每个对象拿出来后，传到item里面，先声明**

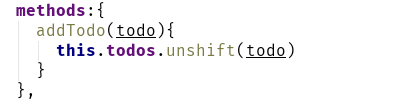
****

**之后在每个li中使用对象里面的属性**

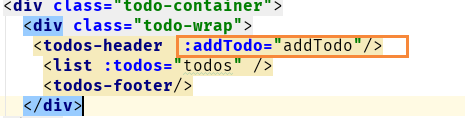
****

**二、添加内容：**

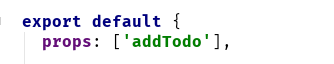
**先在app中定义添加todod的函数**

****

**按照props的方式传入到组件中**

****

**同样要在header组件中声明**

****

**然后收集input的输入值以及给input添加事件监听**

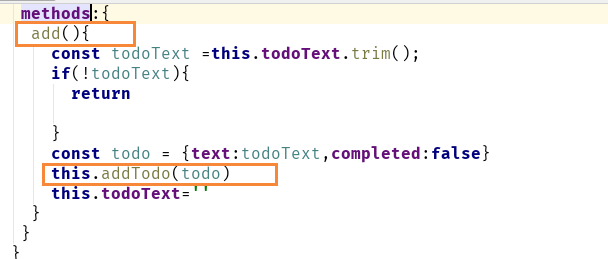
****

**并在data中初始化**

****

**写事件监听函数**

**通过keyup的方式然后在函数中接收input中值，然后通过app传过来的函数中添加到app是数据里面**

****

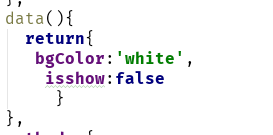
**这个效果**

****

**首先这个是在item中的每个li实现的**

**首先li定义一个style的颜色，初始化为白色，给button一个事件初始化是不显示**

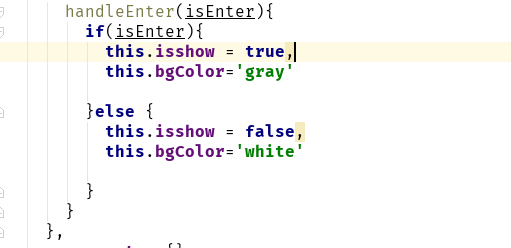
****

****

**然后再给li添加事件监听**

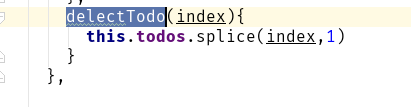
****

**定义相同的函数，传入不同的参数，当它是进入就改变按钮和背景颜色**

****

**实现删除**

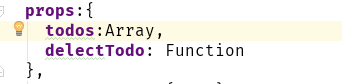
**首先在app中定义删除函数，一个一个的传递**

****

**以属性的方式传入list组件**

****

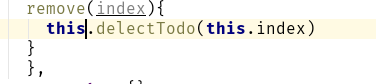
**在list组件中声明**

****

**在list组件传入属性，在继续**

****

**给删除按钮添加点击事件，如下图代码所示**

****

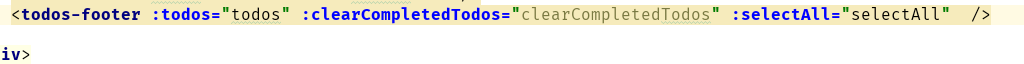
**全选全不选**

**全选**

**首先要明确脚部需要几个值**

1. **首先是要todos的全部有多少项**
2. **当前选中的有多少个**
3. **是否全选**

**首先在App中先完成操作底部需要的数据函数**



**这个todos是当前数据的总长度，传过去**

**这个clearCompletedTodos是todos已经完成，就是这个集合的里面为true的数量聚合**

**这个首先要遍历数据中每个对象的的是否完成**

**定义两个函数，第一个函数是将todo为false，的点击之后就删除**



**给button添加事件监听**



**直接声明之后，就可以直接用了，添加名字上去，就可以将函数都放入**