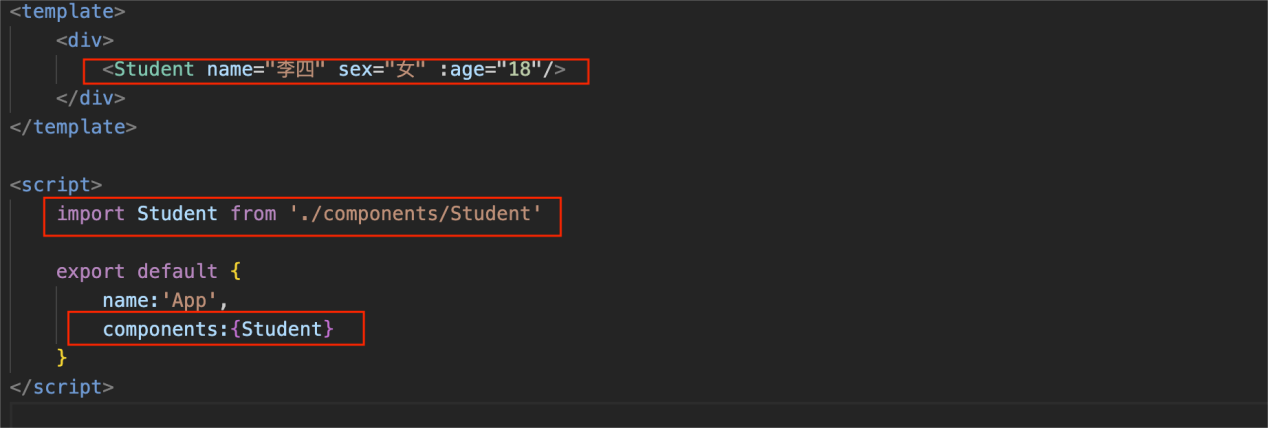
# Vue基础---插槽

在项目的日常开发中，我们不可避免的会饮用一些外部组件或者自定义组件；有了这种引用关系之后，我么就可以把他们称为父组件或者子组件，同时父子组件之间有很多的通信方式，比如可以通过props向子组件传递数据，或者通过$emit、$parent调用父组件中的方法。

在子组件Student组件中，通过props接收来自父组件的数据



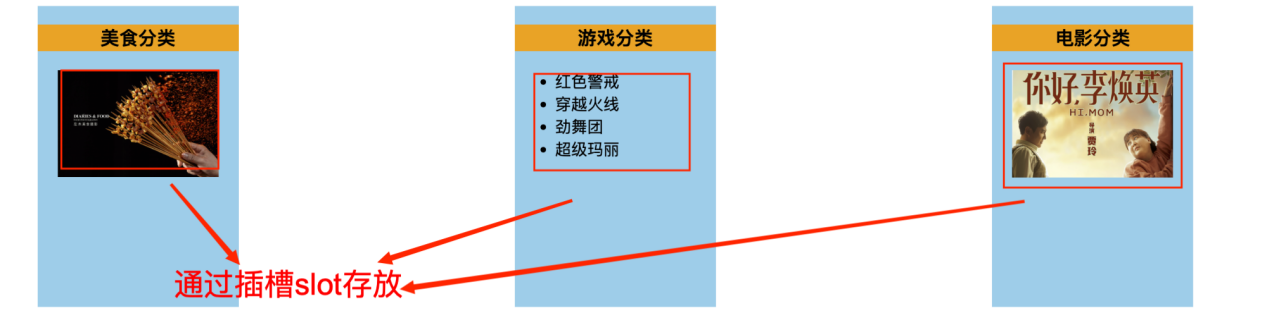
在父组件App中，引用子组件Student



最后运行项目，子组件的内容成功被引用并展示在页面上

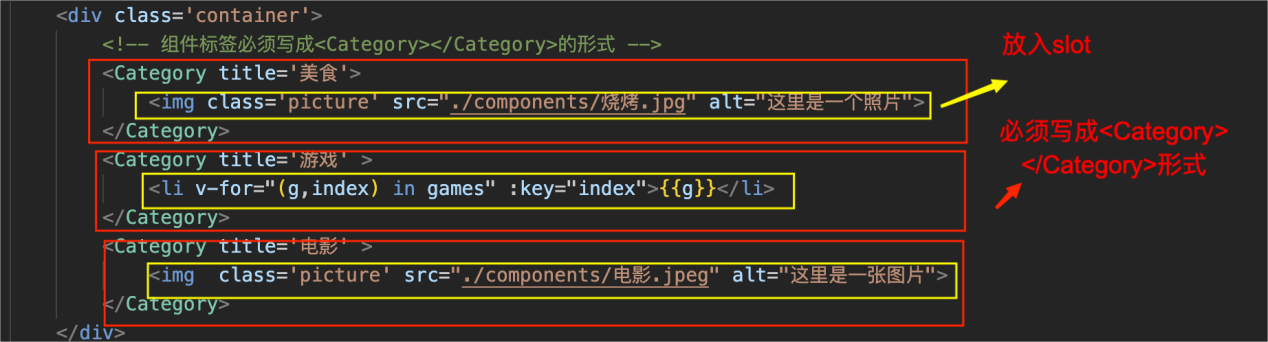
那假如我们现在有这样一个需求：在引用子组件的同时，希望在子组件指定位置插入一段内容，我们直接在父组件App中的<Student></Student>标签中插入是不起作用的，为了解决此类问题，vue设计出了slot这个东西，slot翻译过来叫做插槽，也称其为vue的内容分发机制，它的主要作用是 向子组件的指定位置插入一段内容，这个内容可以是Html或者其他组件。

#### 默认插槽：



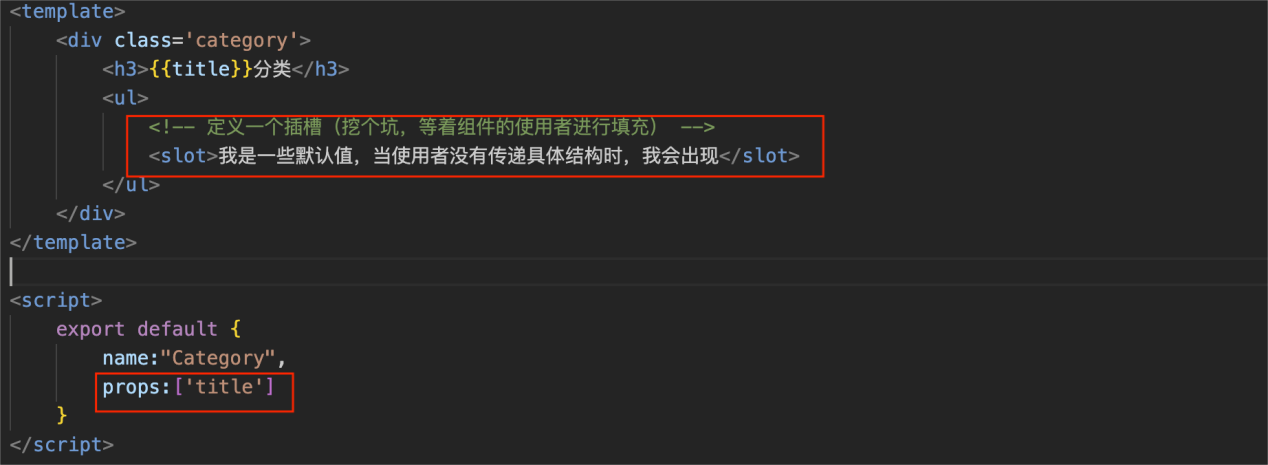
App组件中：

把图片或者li放入 对应的Category组件的slot位置



Category组件中：子组件中写入slot

通过Category组件中存放dom对象，然后传到slot插槽中

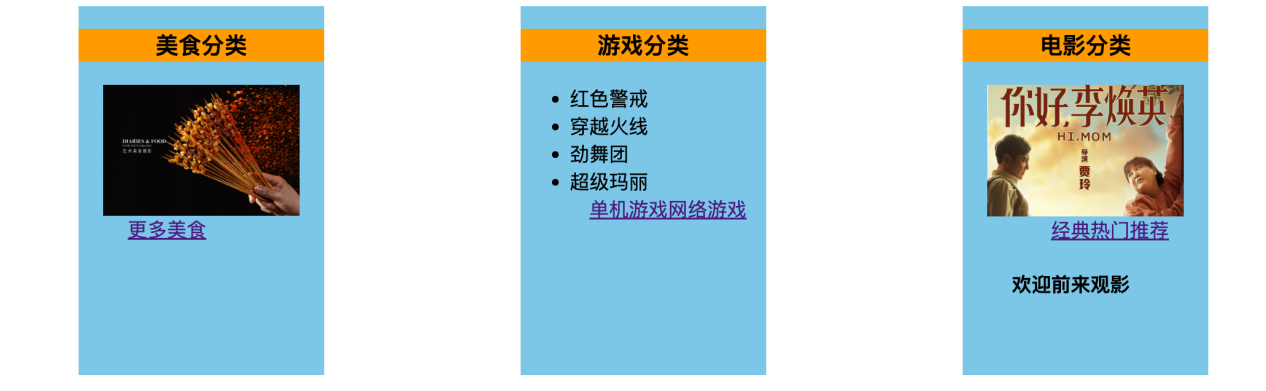


【注意】：可以在App中设置图片的样式，也可以在Category组件中设计样式

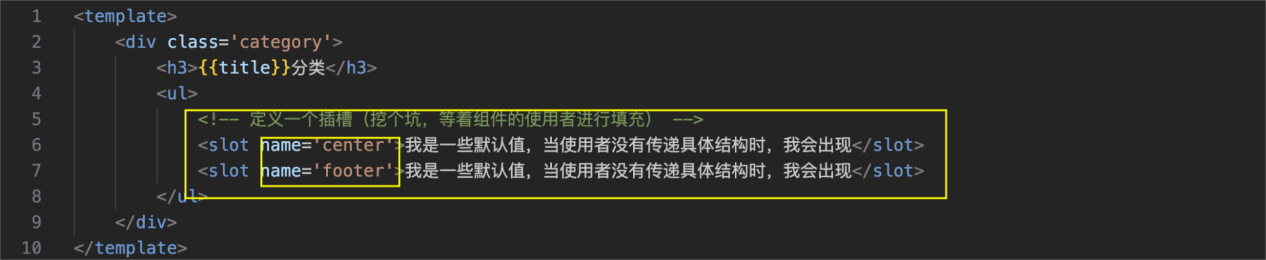
#### 具名插槽：

当需要多个插槽的时候

具名插槽就是给我们的插槽起一个名字，即给<slot></slot>定一个name属性



在子组件中，



给插槽起了名称以后，我们在父组件中就可以使用v-slot:name或者#name或者slot:’插槽名称’ 往指定的插槽填充内容。（#name是v-slot：name的简写形式）

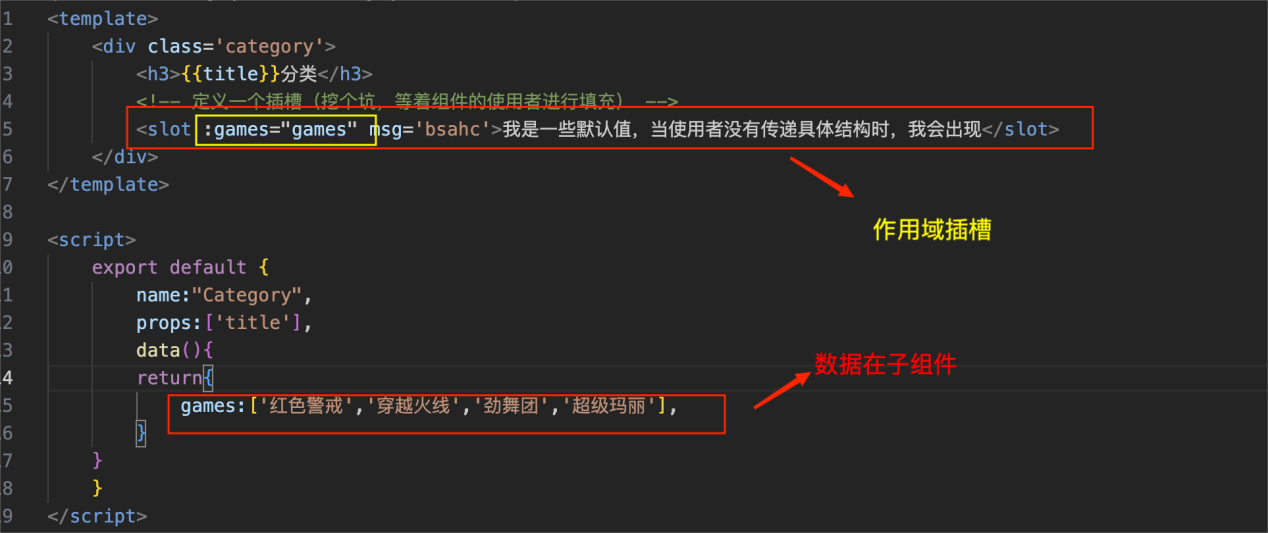
【补充】默认插槽也有name属性，其值为defalut。



#### 作用域插槽

理解：数据在组件的自身，但根据数据生成的结构需要组件的使用者来决定。（games数据在Category组件中，但使用数据所遍历出来的结构由App组件决定）

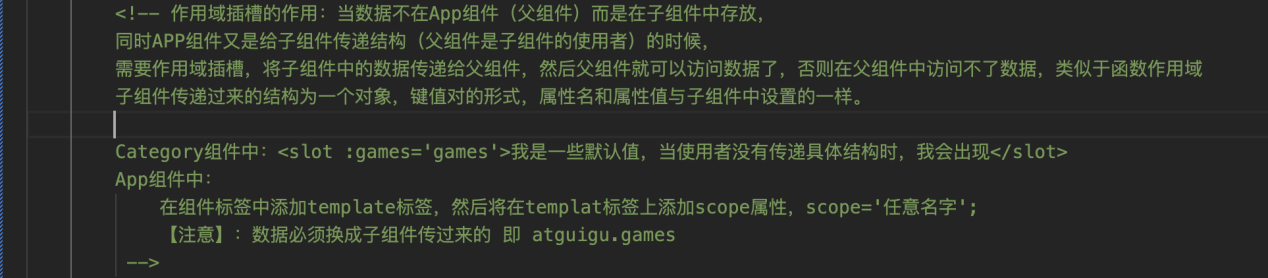
Category组件



App组件



自己的理解 ：作用域插槽有点儿类似于子组件给父组件传递数据



参考博客：[https://juejin.cn/post/7208015274079469627#comment](https://juejin.cn/post/7208015274079469627" \l "comment)