

第二课 组件化的含义

学习先导介绍

- 为什么要有组件化
- 组件化的含义

一. 为什么要有组件化

面向过程开发的缺点

在正常开发一套网站或者应用的时候，很多时候总体需求和原型图设计是相对明确的，主题业务功能和业务流程是明确的，但是细节性的功能点设计往往是并非完全确定。

绝大多数情况下需求的变更往往会对前端开发人员带来较大的影响，比如说好让你做一个弹窗提示用户一段文本就可以了，可能过段时间，需求又变更了，需要你做的弹窗要有标题栏和底部操作栏，你刚一按照要求改好，又需要你把弹窗内容由一段文本变成可以操作的表单等等，这种在开发过程中不断变更的需求和页面调整是非常让人头疼的。

传统的面向过程的开发就是用的这种思路在工作，这种模式是以什么正在发生的功能需求为主要目标进行编程，你也可以理解成这是一种写到哪想到哪在开发到哪的工作模式。可想而知，当做的页面多了，需求变更的多了，开发人员多了，写的东西自然就会乱。

因此我们需要一套统一的开发标准或者说是通用核心功能封装，也可以说是通用工具封装，因此组件化的概念就被提出了，也越来越受到大多数公司的重视。组件化的开发模式能够保证大家都有统一的开发流程，都能够做出统一的开发组件，也能够用同样的参数来调用别人做好的符合要求的组件。

面向对象组件化开发的优点

1. 拆分公用代码
2. 封装单一功能
3. 单独维护

4. 可复用、可继承、可扩展
5. 多态（同样的操作在不同使用者身上会有不同的体现）
6. 从整体角度出发

因此组件化的开发模式可以快速搭建并完成页面，保证代码简洁与可维护，避免反复修改的带来的重复劳动，有利于多人协作共同开发和维护整套项目。

二. 组件化的含义

UI 组件化

这是一种针对页面功能组件的组件化开发模式，一套 ui 组件要包含 HTML、CSS、JavaScript，其中要有这个组件的结构、样式和交互操作等封装内容，外部使用者只需要按照组件提供的参数、方法、自定义时间就可以调用，而不用考虑 ui 组件的内部编写逻辑和业务流程，这其实对于快速搭建页面是特别有用的。

功能组件化

与 ui 组件化相对应的是功能组件化，这里的组件化封装主要是针对一些业务核心功能的封装，比如说封装一套 ajax 进行 http 请求的功能组件，我们在这个功能组件的内部已经配置好了相关的请求头信息、请求拦截设置以及请求反馈函数等相关操作功能，使用者只需要按照一定的规则，通过传递对应的参数和回调函数就能调用我们封装好的请求功能。

谢谢观看！

我是星星课堂老师：周小周