

## 第十六课 js 设计原则

### 学习目录

- 单一职责原则
- 开放封闭原则
- 最少知识原则

### 一. 单一职责原则

应该有且仅有一个原因引起类或者对象的变化。假如一个类或者对象承担了过多的职责，并且与其他类或者对象有太多的联系，那么在需求的改变过程中，需要改写这个类或者对象所带来的麻烦就越多。

在 js 中单一职责原则主要体现在了对象与函数的封装，如果一个对象或者函数的内部只为了完成某一项工作，那这个对象和函数就是符合单一职责原则的。

#### 优点

- 1.分离代码功能逻辑，让代码粒度更细，代码层次结构更加清晰。
- 2.有利于代码复用，有利于测试问题排查。

#### 缺点

- 1.增加了封装代码编写的复杂度。
- 2.代码的联系性变差，增加对象函数之间相互联系的复杂性。

#### 何时采用单一职责原则？

- 1.单一职责的划分一定是有层次的，某几个符合单一职责原则的对象或者函数一定能够

组合成另外一个层次的对象或者函数。

2.如果在一个对象或者方法中有两个或多个职责都会同时变化, 那么不需要使用单一职责原则。

3.也可以在代码编写进入调试优化的时候在考虑使用单一职责原则进行优化改写。

## 二. 开放封闭原则

已经定义好的对象或者函数需要修改或者增加新功能时, 可以增加新代码进行扩展, 但不可以修改这个对象或者函数。这个原则的核心思想是如果要扩展那是开放的, 随便扩展, 如果要修改那是封闭的, 不给修改。

### 优点

- 1.保证原有对象或函数的稳定性。
- 2.代码属于抽象编程, 用抽象分离容易变化的部分。

### 缺点

- 1.很难完全做到纯开放封闭, 代码的维护成本也较大。

### 何时采用开放封闭原则?

1.一个对象或者函数中有大量逻辑条件语句时, 这意味着之后可能会增加新的条件, 所以适合运用开放封闭原则把变化的逻辑田间语句抽象出来形成一个类或者对象统一封装与管理。

## 三. 最少知识原则

一个变量、对象、函数等要尽量少的与其他变量、对象、函数相互作用。其实就是某几个变量、对象、函数只和与他们相关的业务逻辑联系, 不和不相关的对象打交道, 如果要和

不相关的对象打交道，可以通过一个第三方对象来打交道。

假如两个人互相不认识对方，这两个人如果要发生交流，可以通过第三个人作为第三方来相互交流沟通。

### **优点**

1. 减少了变量、对象、函数等的相互耦合，让代码的功能性更加明确，结构更加稳定。

### **缺点**

- 1.需要编写和维护复杂的第三方对象，增加开发难度。

### **何时采用最少知识则？**

- 1.面向接口编程时，使用者不需要知道封装对象的实现逻辑，只需要通过一个方法入口就可以使用封装对象的方法。

- 2.一些对象或者函数需要相互关联通信的时候，可以使用最少知识原则来构建满足需求的第三方对象。

## **谢谢观看！**

我是星星课堂老师：周小周