Get in touch at
www.ccjk.com/contact
浅谈设计模式
设计模式，英文名为 Design pattern，它起源于建筑学，由Erich Gamma等人在1990年代从建筑设计领域引入
到计算机科学的。它是对软件设计中普遍存在也是反复出现的各种问题，所提出的软件解决方案。
经过这么久的编码，让我越来越感觉到设计模式的重要性，设计模式于己于他人于系统都是多赢的，设计模式使代码编制真正工程化，设计
模式是软件工程的基石，如同大厦的一块块砖石一样，要是砌得不好，仅仅是想改变一下大厦房中的格局，就要把整栋大厦推倒重建，当然，
这是万万不可的。
如今的设计模式是一套被反复使用、多数人知晓的、经过分类编目的、代码设计经验的总结，使用设计模式可重用代码、让代码更容易被他
人理解、保证代码可靠性。一个好的设计，往往会包含各种各样的设计模式，它就像一个可以活动的大厦一样，自由转动大厦里面的各个房
间，改变他们的格局，而不会让大厦倒塌。
现在简单介绍一下面向对象的几个原则，为什么会有这些原则？根本原因就是为了代码复用，增加可维护性。其中一项开闭原则具有理想主
义的色彩，它是面向对象设计的终极目标。而设计模式就是实现了这些原则，从而达到代码复用、增加可维护性的目的。
开闭原则：原文是：”Software entities should be open for extension,but closed for
modification”。就是说模块应对扩展开放，而对修改关闭。
里氏代换原则： 如果调用的是父类的话，那么换成子类也完全可以运行，反之则不成立。
依赖倒转原则：抽象不应该依赖于细节，细节应当依赖于抽象，要针对接口编程，而不是针对实现编程。即是抽象就像大的解决方案，细节
是实施的具体方法。
接口隔离原则：定制服务的例子，每一个接口应该是一种角色，不多不少，不干不该干的事，该干的事都要干。
遵循着这些原则，设计模式已经被人们发展出最为常用的23种设计模式，这些模式已经非常成熟，得到大多数人的认可。目前我也在学习，
可以说能够掌握的也就只有四种模式而已，但仅这四种也为我带来了很多思想，灵感，往后我将我掌握的这四种模式会与大家分享。
最后推荐一部好书：《软件秘笈：设计模式那点事》