

Java命令模式:设计模式的四人帮计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_Java_E5_91_BD_E4_BB_A4_c97_645321.htm Command 命令模式 Intent :

Encapsulate a request as an object , thereby letting you parameterize clients with different requests, queue or log requests, and support undoable operations. 瞎谈：其实很好理解。命令模式，关心的就是命令(或者称为操作)。打个比方。在一个公司里面，整个运作就像一个系统。某个boss发布了一个命令，中层领导接到这个命令，然后指派给具体负责这个员工。整个流程很清晰吧。有一个需求，如何将这个流程固定下来，形成一个系统。我们只要抓住了重点：命令。将它抽取出来，其他的都迎刃而解了。抽取命令，封装成一个独立的对象，实现了解耦。至于其他的，可以方便地扩展，不论这个命令是CEO，人事部，还是你爸提出来的。无论这个命令的执行者是张三还是王八。这个模式的产生，其实是哲学上的“抓住主要矛盾”。更多的例子，如其他作家举的“去路边吃烤肉和去烤肉店吃烤肉有什么不同”或者“美猴王大闹天宫中玉帝拍太白金星捉来猴子”。正经：命令模式把一个请求或者操作封装到一个对象中。命令模式运行系统使用不同的请求把客户端参数化，对请求排队或者记录请求日志，可以提供命令的撤销和恢复功能。Java命令模式本质是对命令的封装，从而把发出命令的责任和执行命令的责任分割开了，委派给不同的对象。通俗地说，我是老总，我只管发个命令，至于这个命令发给谁，谁执行，关我P事，我发钱请人不是为了给自己找麻烦。你是负责事情的员工，你的天职是做好上

级交给你的任务，踏踏实实，不要知道太多，不要八卦，不要问太多了。好处：很容易构造一个命令队列 记录相关的命令日志 增加命令的状态，实现命令的撤销和重做 允许接受请求的一方决定是否可做 新的命令轻而易举可以加入其中 缺点：可能会有过多的具体命令类存在 实现：也不难，第一步关键是建立Command对象。拥有一点面向对象的思想，就知道把它先抽象，让继承它的对象去具体实现。Client对象是发布命令的。Invoker对象是传递命令的，就是跑腿的。Receiver是受气包，底层最累的程序员，负责干活吧。看看下面的类图就清晰了：上面的类图，一开始我有一些疑问，不如Invoker为什么要存在。现在想通了，Client的职责只是发布命令，就不要给它增加传递命令的职责。因为客户有千千万万。在现实中，跑腿的人为什么要存在呢？因为领导之所以是领导，就是只发表命令，跑腿等差事就给跑腿之人去办吧。现实中，跑腿之人通常比做事之人混得要好些。因为他们之间面对的是领导，把做事之人功劳揽到自己身上。唉，没想到设计模式蕴含了如此深刻的道理，佩服“设计模式的四人帮 Gof” 文章来源:百考试题网 Java命令模式实现例子：阎宏博士的书中举了很多生动有趣的例子，大家可以去查阅。比如一个Mp3。你按了一个播放键盘，就播放了。这就可以算是命令模式的一种。你是Client，按键是Invoker，mp3是Receiver，播放就是一个命令Command对象。编辑特别推荐: java面向对象编程笔记:数组的相关事项 Java基础学习中必须清楚的一些问题 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com