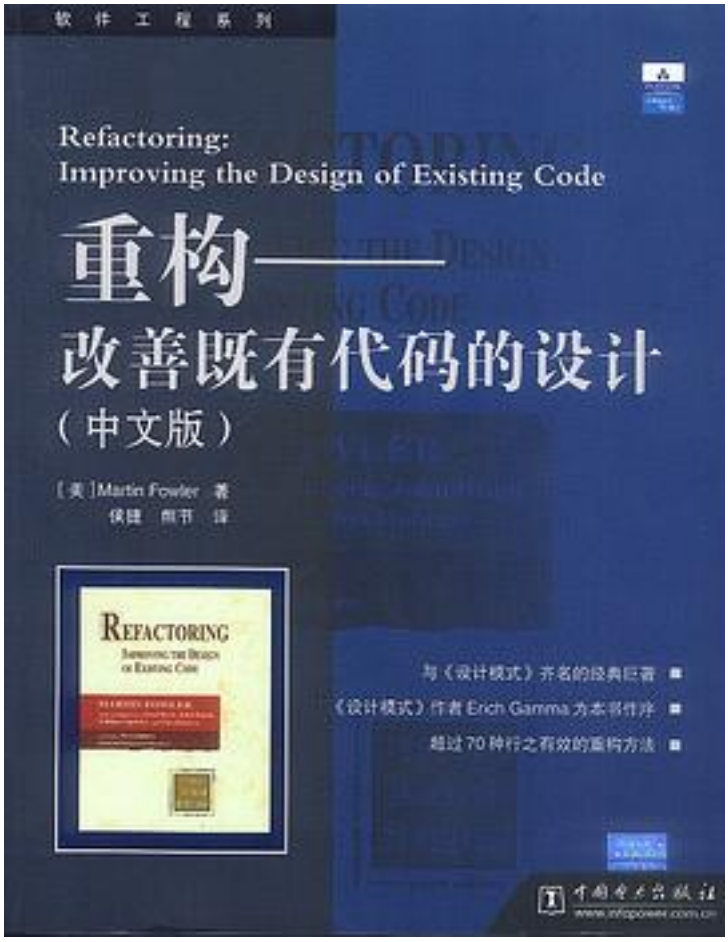
# 重构




重构_下载链接1_

著者:[美]马丁·福勒（Martin Fowler）

出版者:人民邮电出版社

出版时间:2015-8

装帧:平装

isbn:9787115369093

本书清晰揭示了重构的过程，解释了重构的原理和最佳实践方式，并给出了何时以及何地应该开始挖掘代码以求改善。书中给出了70
多个可行的重构，每个重构都介绍了一种经过验证的代码变换手法的动机和技术。本书

提出的重构准则将帮助你一次一小步地修改你的代码，从而减少了开发过程中的风险。

作者介绍:

作者介绍

Martin
Fowler，世界软件开发大师，在面向对象分析设计、UML、模式、XP和重构等领域都有卓越贡献，现为著名软件开发咨询公司ThoughtWorks的首席科学家。他的多部著作《分析模式》、《UML精粹》和《企业应用架构模式》等都已经成为脍炙人口的经典。

译者介绍

熊节，ThoughtWorks中国公司的高级咨询师、架构师和项目经理，在大型企业应用及互联网应用的架构和管理方面拥有丰富经验。作为敏捷方法学顾问和重构专家，他拥有在各种技术平台、编程语言、软件形态的项目中实施重构的丰富经验，并曾主持极具挑战性的超大规模电信软件系列重构工作。

目录: 第1章 重构，第一个案例 1

・・・・・・(收起)

重构_下载链接1_

## 标签

编程

重构

计算机

软件工程

程序设计

重构与设计模式

软件开发

编程自我提升

## 评论

书中介绍的重构技术都十分实用，讲解过程也十分精炼。程序员进阶必读的书籍

---
很好的一本书，感觉随着开发经验越多，读起来理解会更深刻，往往会发现产生了共鸣，自己很多地方也是这么做的

---
经典之作，读起来轻快。第二章、第三章是精华。多实践，多做 Code Reivew。

---
虽然以Java作为本书讲解的语言，没用过java，不过有C++这样的面向对象语言技能也能看懂大部分内容。

---
1.1

---
关于重构，很赞成Kent
Beck在结尾写的话：这有点像在悬崖峭壁上的小径行走，只要有光，你就可以前进，虽然谨慎却仍然自信。但是，一旦太阳下山，你就应该停止前进；夜晚你应该睡觉，并且相信明天早晨太阳仍旧升起。

---
很多重构方法，可以借鉴参考。

---
读了三遍了，以后再读就读新版的了

---
对于学习面向对象编程很有帮助，perfect。

---
启发好多呀，曾经读过，一头雾水。现在读来，才觉得好多东西，真的可以用在实际中了。提升不是一星半点，而且一种思维。当然，读它，你至少有一些Java的基础。很好，值得阅读

---
简单过了一下，具体手法没怎么看。加深了对重构这个技术的认识，想进行更多的尝试。

---

重构经典，作者的语言也很风趣幽默。

---
重构的意识与尺寸比方法论重要，需要大量有意识的去思考这里需不需要改，为什么要改，怎么改。重构也可以说对设计模式的另一方面的运用。

---
程序员保命神书!

---
算是进阶类的书籍，对于编程的自我提升有很大的作用，读了一遍确实受益匪浅，还是需要多度多理解。

---
因为用的java 暂时只看了偏理论的部分 /
很奇怪，这本15年版的比10年版分低那么多（7.8-9.0 / 人数10+ - 1000+）

---
重构_下载链接1_

## 书评

2009年，在为《重构》第一版的中译本再版整理译稿时，我已经隐约察觉行业中对“重构”这个概念的矛盾张力。一方面，在这个“VUCA”（易变、不确定、复杂、模糊）横行的年代，有能力调整系统的内部结构，使其更具长期生命力，这是一个令人神往的期许。另一方面，重构的扎实工夫...

---
纵览武侠江湖，制胜法门不外两项，内功和外功。二者得一可天下去得，但最终皆入内外兼修之境
倚天是自内而外，先修内功九阳真经，然后以此为基础，加上太极拳和太极剑，最终成就天下第一高手 笑傲是自外而内，先学独孤九剑，后学吸星大法，最后学易筋经。
神雕也不外如是，玉女...

---
书中说过重构的思想由来已久，只不过没人写成书籍罢了。

个人感觉如果你看完这本书只看到了思想，那你就错过了很多有价值的东西。小步骤的重构，如何最大限度的不引入bug才是书中要告诉大家的。像了解重构思想，只看前几章就够了。我个人认为书中最有价值的东西恰恰是进行重构...

---
整体通俗易懂, 翻译过程有些小错, 不知道有没有勘误表 (1)184页下面范例跳了好几步 (2)187页范例, 数组说三个元素,代码只列出两个, "失败场次"没有了, (3)282页, 的代码sentAlert()函数好像有问题, 变成递归了. (4)306页, Raname Method明显是印刷错误了.

---
代码的坏味道章节描述了常见的不良代码，而且基本上流行于各种编程语言。
第6~11章概括了一些java中的重构手法，是作者在实际操作过程中重构笔记的基础上总结而成，对于我们非Java程序员来说选择性阅读即可。
重构手法中包含了不少譬如提炼函数和内联函数这样的相反的操作，这...

---
重构是个好的思想，第一次读此书是在大学里，张忠强介绍给我。后来在大学做项目的时候，每每遇到困难难以进行的时候，我就开始重构我的代码。现在到了公司，以C语言为主要开发语言，我同样在运用着重构的思想在工作，这次添加新功能完全是一边重构一边完成下来的。重构在我看来...

---
组里最主要的Service已经运行了几年了，目前大约有40000行代码，不少部分缺乏Unit Tests。每次看代码的时候都有一种想重构的冲动。
不过什么时候才重构呢？经理那里是不好交差的——他们关心的是新功能的实现速度。有的时候重写反而（对程序员）的发展更好，因为工作量明显的...

---
适合写过两年代码的开发人员的一本书。
示范代码难度不是很高，主要就是理解里面的一种精神了。
我才看了一章，希望能坚持看完。

---
这本书一开始读的是英文，不过Martin Fowler本身好像就不是一个Native的英语国家的人。所以他的英文写的也是比较容易懂的。
这个书第一章是一个完整的重构的例子，虽然现实中不可能如此完美而孜孜不倦的重构，不过作为一个例子，是非常的好的，让你能为重构的力量所震...

---
《重构》第二版来了！
很高兴有机会能够提前阅读这本神奇的书籍！很多年前就已经阅读过第一版《重构》！那时候就给我很大的冲击！说实话，开始并没有对这一版抱有太大的希望！但是当翻开书卷的那一刻，我还是很庆幸没有因为自己的执念而放弃掉这样一本真心重构了的好书！ 可以...

---

---
无论你是初出江湖的编码小农，还是深耕多年的程序大牛，这本书都值得你深度品读。常常我们说的代码简洁性、易读性、健壮性，都并非一日之功，需要在日常的迭代中不断持续的进行重构，重构的事情我们常常挂在嘴边，却往往无从下手，本书以作者丰富的编程经验和思想，总结了一套...

---
《重构》这本书我觉得没什么意思。这本书有个矛盾点：对有丰富经验的程序员来说，这本书上面所提到的所谓手法已经是他们日常编程中已经熟练使用的，这本书只是给这些手法命了个名而已，看这本书毫无意义。而对于新手程序员来说，因为你没有一定的代码量，应该是看不懂书上说的...

---
第一遍于两年前, 让我了解到bad smell们, 还有那冗长的手法catalog. 最近第二遍,
回顾过去, 虽然bad smell的提示和重构操作已经完全融入现代IDE中了,
但总结下来它对我的影响还有如下: 1.我常常通过重构来加深对代码的理解.
2.让我变成一个十足的代码洁癖者, "刻"代码上瘾...

---
《重构》是一本简单实用的好书，每个靠写代码领工资的软件工程师都应该读一读。运用重构技术可以帮你写出更好的代码——这会让你和你同事在阅读、修改代码时轻松很多。 大学毕业后我用vim + C语言工作一年多，Visual Studio +
C++工作两年半，现在用Eclipse + Java工作了一年...

---
无容置疑，这是一本经典的书，正如推荐所说，是这本书让重构这个看似高深莫测的话题走进了寻常程序员的世界。
可是我们不得不注意这本书的出版日期，是七八年前，和设计模式一样，当年的设计模式那是高手的工具，开口闭口带那么一两句，工厂，单例，会让人对你崇拜至死。可是...

---
纸张很赞，只是排版松散，书就比较厚了。
本人对代码有洁癖，不自觉的一直会调整过去的代码，也需要对照大师总结的规律反思一下。 重构的基础能力在于能够嗅出代码中的坏味道（Bad smell），甚至反模式（Anti-pattern），因此要结合相关书籍一起阅读了。
最近在看的肖鹏一篇文...

---
《重构》给我最大的收获，其实是想法而非技巧。
开发人员的完美主义倾向，容易变成他们做Big Design Up Front的驱动力。而在现实的环境中，时间、资源等约束条件，通常不会允许我们进行过于详细的设计。 我认为最好的做法是，轻量级的设计 + 适当的重构，迭代地开发出满足各...

---
第一次看有网友极力推荐这本书，就买了一本英文的来看，一下子就被吸引住了。
原来觉得编码可以天马行空，总想看看那些高手是怎么写代码的，现在觉得自己好像也是高手了，至少可以看得出哪些代码好，哪些代码差。真是感谢Martin Flower。
现在自己做项目经理了，对于新来的有一...

---
重构是设计,设计是art,重构也是art.
一个函数三行只是语不惊人死不休的说法,是对成百上千行代码的矫枉过正。
更一个般的看法是一个函数应该写在一页纸内。但举一个上百行的极端例子也是可能的。比如某种数值计算。
重构的意义应该在XP的背景下看，如果没有重构，XP的方法从...

---
重构_下载链接1_