<<构建高质量的C#代码>>

图书基本信息

书名：<<构建高质量的C#代码>>

13位ISBN编号：9787121197130

10位ISBN编号：7121197138

出版时间：2013-4

出版时间：电子工业出版社

作者：曹化宇

前言

本书关注的主题是如何使用C＃写出高质量的代码。
我们知道，高质量的代码是优秀软件的基础，创建高质量的代码是每一个真正的程序员坚持不懈的追求，同时，也是一项艰巨而又充满乐趣的工作。
我们认为，在编写高质量代码的过程中，需要考虑以下三个方面：编写高质量的基础代码。
我们会使用某种编程语言写出语句或基础结构，比如我们使用C＃语言定义变量，或者是创建选择语句、循环语句结构，又或者定义方法、结构、类、接口，等等。
这些内容作为软件代码中的基本元素，与所使用的编程语言的特点是息息相关的。
只有当我们熟练掌握并合理使用了编程语言的特点，才能写出高质量的基础代码。
创建高质量的代码结构。
在这里所说的代码结构，是指在软件开发中，用于实现业务模型的软件架构。
高质量的软件架构可以帮助我们有效完成业务模型所要达到的设计目标，同时，也可以应对不断变化的软件需求。
在软件开发和经验积累的过程中，产生了众多的软件开发技术和方法，它们可能与编程语言无关，但对应于每一种编程语言，实现起来又有一定的独特性。
设计模式就是在不断的开发和积累过程中，发现并总结出的一系列用于优化代码结构的方法。
在软件开发中合理地使用成熟的设计模式，可以帮助我们高效地创建灵活、易扩展、富有弹性的软件架构。
这也是我们在应对“需求不断变化”这一软件开发中唯一不变真理时的常用且行之有效的开发技术。
逐步改进。
无论是基础代码还是代码结构，要想在初次开发时就能达到很高质量的可能性并不是很大。
因此，这就需要我们对代码进行不断的改进，而改进的步骤应该是循序渐进的，而不是暴风骤雨似的。
在软件开发中，代码和架构的改进过程，也就是重构（Refactoring）的过程，需要我们对每一条语句或语句结构、算法、架构进行逐步的改进和优化。
其中，对于软件架构的重构过程，往往就是应用或去除设计模式的过程。
而我们应该注意的是，重构是在不改变或少量修改代码外在行为的基础上，对其内部结构的优化，以便达到代码和架构设计更合理、更简洁、更灵活的目的。
重要的是，只有在不断的尝试和挫折中才能让我们积累到更多的、有价值的软件开发经验。
此外，我们应该注意代码逐步改进的过程，在这个不断尝试的过程中，会产生多个代码和架构的版本，只有通过对比，我们才能找出最合适的解决方案，这也是经验积累的重要过程。
本书将结合C＃编程语言、重构与设计模式，在掌握C＃语言基础知识的同时，进一步学习如何使用C＃语言实现一些常用的设计模式，而重构则是不断地进行基础代码与软件架构逐步优化的过程，在这一过程中，我们将需要更多地理解高质量代码的进化过程，从而可以帮助我们在使用C＃语言开发软件时，能够更合理、更高效地创建出高质量的代码，并使软件架构变得更易维护、更灵活、更富有弹性。
本书特点享受充满乐趣的学习与开发过程。
软件开发应该是一个充满乐趣的过程，而软件开发的学习过程也应该是这样。
本书使用了通俗易懂的语言，结合比较有趣的示例代码，比如大量使用了模拟创建游戏的代码示例，让大家在一个比较轻松的氛围中学习和应用知识点。
丰富而实用的代码示例。
本书包含了大量的示例代码，并在其中穿插了很多实用的内容，比如随机排序算法、中国农历信息的获取、树状结构，以及大量的设计模式模型，等等。
大家可以在自己的软件中直接使用这些代码，也可以根据需求对其进行修改、扩展或者简化，从而可以更有效、更合理地运用这些算法和代码结构。
大量实用的开发技巧和方法。

在讨论知识点的时候，我们列举了大量的实用开发技巧和方法，并通过实际测试来解答一些技术上的疑问，真正做到能见到树木，亦能见到森林。
本书内容本书主要包括以下几个部分的内容：C＃编程语言，涵盖了使用C＃开发应用软件过程中常用的内容，如数据类型、结构化语句、数组、结构类型、枚举类型、类、委托、泛型、接口、多线程与资源同步等。
并对一些不太常用的内容做了简单的介绍，如预处理、特性、不安全代码、分部类型和可空类型等。
设计模式，介绍了在C＃和．NETFramework中应用的四种设计模式，包括访问者、迭代器、适配器和模板方法模式。
然后，我们详细介绍了11种设计模式，它们是策略、工厂方法、抽象工厂、生成器、单件、观察者、享元、组合、装饰者、状态和桥接模式，并创建了这些设计模式的基本模型，这些都是在C＃开发应用软件过程中可能会经常使用到的模式。
读者应该在学习和使用设计模式的过程中领悟到软件开发的精髓和本质所在，而不只是照葫芦画瓢。
读者应该明白，应用设计模式并不是我们的目标，我们的目标是创建高质量的软件，任何技术的取舍都应该基于这一目标。
在贯穿全书的内容中，我们穿插介绍了各种编写高质量代码、架构，以及逐步改进的方法、技巧，并给出了一些代码演化的过程。
最后，我们讨论了C＃、设计模式、重构，以及软件构建中涉及的一些主题，感兴趣的读者可以选择一些主题进行深入的学习和研究，以扩展自己在软件开发方面的知识面。
读者类型对于本书的读者，你可以是：软件开发的初学者，可以从基础的C＃编程语言开始学习，然后，逐步学习设计模式的应用过程，以提高自己对代码结构整体的把握能力。
从其他语言转到C＃语言的读者，无论你是不是已经会使用VB、C＋＋，或者Java，都可以很快地适应C＃开发的便利性，并且可以通过设计模式在C＃中的应用，更深入地理解C＃语言的特点和软件构建方法。
C＃使用者，如果你正在使用C＃语言进行软件的开发工作，可以从本书学习到如何在C＃中应用一些常见的设计模式，以便提高我们对软件代码的重构和优化能力。
同时，从C＃编程语言的角度，本书也可以作为参考手册，供你随时查询相关的知识点。
本书的使用如果你对C＃语言和．NETFramework不太熟悉，可以从第1章开始逐章阅读，一步步深入学习相关知识。
如果你已经对C＃比较熟悉，可以直接阅读在C＃中使用设计模式的相关内容。
本书也可以作为C＃语言以及设计模式的参考手册，读者在开发的过程中，可以随时在本书中查询C＃语言与应用设计模式，以及了解如何改进和优化软件代码和架构。
此外，代码与架构重构的过程介绍，穿插于不同的主题之中，方便读者直观地了解应用相关技术需要注意的问题，以及代码和架构优化，即逐步改进的过程。
学习方法你可以先通读一遍。
如果你的时间比较紧，至少应该先看一下目录，了解一下本书有哪些内容，这样，可以做到对全书的结构和内容有一个初步的了解。
然后，你可以从第1章开始，或者选择自己感兴趣的内容阅读，但你应该充分理解每一个知识点和示例。
学习软件开发最好的方法就是实践，读者应该多读代码、多写代码。
建议有条件的读者可以自己输入本书中的所有示例，并尽可能地去尝试修改代码，测试不同的方式带来的运行结果上的差异。
相信这么多的代码敲下来，读者对代码的理解会更深入。
另外，这样做还会有一个额外的收获，那就是你的打字速度可能会有所提高。
在阅读过程中，对于不太明白的地方，不用着急，当你系统地学习后，一些难题就会迎刃而解。
请注意，如果你对哪个知识点暂时不能理解，可以先放松一下，如玩玩休闲小游戏、睡一觉，或者出去散散步，再静下心来时，答案就很可能自然地出现在脑海里了。
最后一点，在学习的过程，手边一定要有笔和纸，随便写写、画画对我们的学习都会有所帮助，有时

还会有意外的收获。
交流与反馈就像开发软件一样，撰写一本书也不太可能在一开始就达到完美的境地。
如果读者发现什么问题，或者有些什么建议，又或者只是交流一下软件开发的心得，都可以给我发来E-mail。
祝大家在软件开发的过程中玩得开心！
作者2013年1月

<<构建高质量的C#代码>>

内容概要

《构建高质量的C#代码》介绍了高质量C#代码的成就过程，即从基础代码到软件结构，以及不断优化和重构的过程。
主要内容包括C#的基本语法、结构与应用特点，常用开发资源与技术要点，设计模式在C#中的应用等，以及特别重要的一点，即我们应该养成良好的开发习惯，不仅要注重技术细节，还要从更宽阔的视野角度来重新审视代码的构建工作。

作者简介

曹化宇，独立软件开发人，主要研究方向为软件工程、设计模式与交互设计；最大的兴趣就是coding、技术写作、文学创作，以及研究各种软件开发技术。
在知识的海洋里，每个人的力量如水滴一般微不足道，但我们可以努力去做好自己的事情，从而共同创建一片更加广阔而美好的知识海洋。

<<构建高质量的C#代码>>

书籍目录

章节摘录

编辑推荐

《构建高质量的C#代码》编辑推荐：431段精选代码助您创建更灵活更富弹性的软件架构。

<<构建高质量的C#代码>>

版权说明