第一图书网, tushu007.com

<<面向对象设计原理与模式>>

图书基本信息

书名：<<面向对象设计原理与模式>>

13位ISBN编号：9787302196716

10位ISBN编号：7302196710

出版时间：2009-4

出版时间：斯科瑞、腾灵灵、 仲婷 清华大学出版社 (2009-04出版)

作者：斯科瑞

页数：310

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

内容概要

《面向对象设计原理与模式》全面介绍了Java面向对象程序设计的原理和模式，帮助解决Java程序中的设计问题。
此外，该书十分注重Java面向对象程序设计的每个细节，以及继承、方法、类、设计模式等在程序设计中起到的关键作用，浓缩了Java的精华。
在进行理论介绍的同时，《面向对象设计原理与模式》十分重视实践技能的培养，一些较为综合的实例贯穿了相关的知识点，使学生理解并掌握它们在程序设计中的真正用处和在提升程序性能方面的作用。
《面向对象设计原理与模式》不但可作为高校计算机科学专业的教材，还可作为Java程序设计人员的必备参考书。

作者简介

作者：(美国)斯科瑞 译者：腾灵灵 仲婷Dale Skrien，美国Colby大学计算机科学系的教授，研究方向为算法图形理论，面向对象设计，以及计算机音乐。

书籍目录

第1章　面向对象设计与实现的优雅性1.1　存在的问题1.2　软件工程1.3　设计优雅软件的标准1.4 说明1.5 练习题1.6　参考文献第2章　面向对象的基础知识2.1　面向对象编程与非面向对象编程2.1.1　面向对象编程与非面向对象编程简介2.1.2　面向对象语言2.1.3　面向对象编程的优点2.2　Java中的类、对象、变量和方法2.3　插入语：Java中的类方法和类变量2.3.1　类变量及类方法简介2.3.2　Java中的类变量及其使用2.3.3 Java中的类方法及其使用2.3.4 小结2.4　UML类图简介2.5 实现继承2.5.1　特殊化2.5.2 Java中的Obiect父类2.5.3　特殊化的另一种使用2.5.4　泛化2.5.5　Java中的单继承2.6　类型、子类型和接口继承2.6.1　类型2.6.2　多态2.6.3　多态的价值　2.7　接口与抽象类　2.8　动态方法调用　2.9　重载与重写　2.10　控制对方法和数据的访问　2.11 小结　2.12　练习题　2.13　参考文献第3章　优雅性与实现继承3.1 关于继承的四个观点3.1.1　代码重用观点3.1.2　Is-A观点3.1.3　公共接口观点3.1.4　多态观点3.2　代码重用的充分性3.3　代码重用联合Is-A关系的充分性3.4　代码重用、Is-A关系以及公共接口的充分性3.5　Has-A关系和UML关联关系3.6　代码重用、Is-A关系、公共接口以及多态的充分性3.7　使用实现继承的代价3.8　示例：人、女人和男人3.9　示例：绘制多边形3.10　示例：排序3.11　Java中数组的子类化3.12　回顾：继承与引用3.13 小结3.14　练习题3.15　参考文献第4章　优雅性与方法4.1　编码风格和命名约定4.2　方法与分解4.3 内聚方法4.4　结构良好的对象和类不变式4.5 内部文档4.6　外部文档4.7　案例分析：重写Java中的equals方法4.8　案例分析：重写Java中的clone方法4.9　重构4.10　代码优化　……第5章　优雅性和类第6章　Money类的简单案例研究第7章　设计模式介绍第8章　绘图应用程序案例研究第9章　语言解析器案例研究附录A　UML介绍附录B　编码约定和Javadoc注释

章节摘录

第1章　面向对象设计与实现的优雅性对于阅读本书的大学生而言，您很可能已经学习了一些计算机科学的相关课程，其中有一门对于编程的介绍性课程。
经过认真的学习，您一定掌握了至少一门编程语言的基本编程概念。
您学会了诸如循环、赋值和条件语句等语言结构，也学会了编写这些语言结构的一种或多种方法。
您更有可能已经理解了计算机科学并不仅仅是编程而已，事实上，编程仅仅是计算机科学工作者的一种工具。
遗憾的是，因为没有时间讲解计算机科学中如此众多的主题，所以您将无法获悉解决同一问题的各种不同方法的优缺点。
举例来说，在较低的层次上，存在很多种方法来实现一个结构，例如循环（比如，可以使用for循环，while循环或者递归），其中的某些方法相对而言会更加优越。
在高一点的层次上，存在多种方法来实现一个任务（如归并排序或者选择排序），其中某些相比而言更为快速。
再高一点的层次上，存在各种方法来将一个程序划分为诸如类或者方法的模块，其中某些分割的方法会更加合适。
我们来看一个最后那种情况的例子。
请考虑这样一个问题，该问题需要编写一个模块来存储集合，该集合包括一些人以及他们各自的狗的信息。
这个集合需要被保存为散列表、数组还是其他的集合形式？
这个集合是需要存储Person对象还是Person-Dog这样的对象组合。
Dog对象是否需要有实例变量指向其主人？
而狗的主人又是否需要实例变量指向他所拥有的狗的集合？
如果是的话，这又将是一个怎样的集合？
这样的Person对象还需要存储哪些其他数据？
如果这些人是美国公民，是否需要存储他们的社会保障号码？
是否需要将这样的人的集合排序以便于查找？
如果一个人的某条狗死了，是否需要将其从这个集合中删除，或是将其保留为一个没有狗的人？
很自然地，您会疑惑这些问题的答案是否真的很重要。
如果两个设计，一个只有少数几个类和方法，另一个则包含很多类和方法，当这两个设计都能解决问题的时候，具体使用哪个设计是否真的很重要？
类似地，如果所有版本的循环都能正确地完成计算，具体使用哪个版本是否真的很重要？
如果两个算法都能正确计算，哪个运行更快是否真的很重要，尤其是在考虑到每年硬件处理速度飞速增长的情况下？

编辑推荐

本书特色：根据代码“优雅性”讨论设计和实现，使用小型和中型案例分析来介绍设计原理和模式，大部分设计模式均在解决某个问题的背景中引入，在每章的末尾都有大量的各种难度的练习题，便于学生温故而知新。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com