

<<Java程序设计实例与操作>>

图书基本信息

书名：<<Java程序设计实例与操作>>

13位ISBN编号：9787802437395

10位ISBN编号：7802437393

出版时间：2011-6

出版时间：中航出版传媒有限责任公司

作者：丁永卫，谢志伟，高振栋 主编

页数：375

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>

<<Java程序设计实例与操作>>

内容概要

Java是当今最受欢迎的网络编程语言之一，它是一种简单、完全面向对象、解释执行、动态下载、具有多线程能力、可分布访问数据、健壮且安全的新一代语言。

丁永卫、谢志伟、高振栋主编的《Java程序设计实例与操作》以12个项目全面展示了Java语言的风采，其内容包括Java语言特色和Java开发环境介绍，Java的数据类型、常量、变量、运算符和表达式、控制结构等编程基础知识，Java中类、对象、继承、多态、成员变量、成员方法等面向对象程序设计的概念、特点和用法，数组和字符串的声明、创建和用法，异常的概念及其用法，数据输入/输出方法，Java的多线程机制及其用法，图形用户界面开发，以及Java网络和数据库编程基础知识等。

本书具有如下特点：（1）全书内容依据Java的功能和实际用途来安排，并且严格控制每章的篇幅，从而方便教师讲解和学生学习；（2）大部分功能介绍都以“理论+实例+操作”的形式进行，并且所举实例简单、典型、实用，从而便于读者理解所学内容，并能活学活用；（3）将Java的一些使用技巧很好地融入到了书中，从而使本书获得增值；（4）各章都给出了一些精彩的综合实例，便于读者巩固所学知识，并能在实践中应用。

《Java程序设计实例与操作》配有完善的资料包（包括书中全部实例、习题答案和一个优秀的教学课件），非常适合作为高职高专院校以及各类计算机教育培训机构专用教材，也可供Java爱好者参考。

<<Java程序设计实例与操作>>

书籍目录

第1章认识Java

第2章Java语言编程基础

第3章Java面向对象程序设计

第4章类的深入解析

第5章数组和字符串

第6章异常处理

第7章数据输入与输出

第8章Java的多线程机制

第9章图形用户界面开发

第10章Java网络编程入门

第11章Java数据库编程入门

第12章图书管理系统开发

章节摘录

版权页：插图：本章进一步介绍了Java语言中面向对象编程的相关知识，主要包括类的继承和多态的实现方式，以及抽象类和接口的定义与使用方法。

学习本章时，大家应着重掌握如下一些知识：>在使用子类创建对象时，可以利用子类对象名直接引用父类中的成员变量和成员方法，这被称为成员变量和方法的继承。

>如果在子类中重定义了某个父类中的成员变量，则使用子类创建对象时，子类中的方法操作的是子类中的成员变量，而父类中的方法操作的是父类中的成员变量，这被称为成员变量的隐藏。

>如果在子类中重新定义了父类中的某个方法，则利用子类创建对象时，父类中的同名方法将被覆盖。

即无论是子类还是父类，其他方法调用本方法时，实质上调用的都是子类中的方法。

>使用子类创建对象时，父类的无参构造方法总是优先被执行，这被称为构造方法的继承。

当然，我们也可以利用super关键字显式调用父类的其他构造方法：>在Java中，我们可以将基于子类创建的对象声明为父类对象，或者说可以将基于子类创建的对象赋值给父类对象，这对实现程序的多态性非常有用。

>多态性被称为“一个对外接口，多个内在实现方法”，它可以通过方法覆盖和重载方法来实现。其中，所谓重载方法是指：多个方法的方法名相同，但方法参数的个数、类型或顺序有区别，并且返回数据类型也可以不同。

在调用这类方法时，系统会自动依据参数情况来决定调用哪个方法。

>抽象类和接口的主要目的都是为了使程序的功能描述和功能实现相分离，并且一个功能描述可以对应多个实现方法，从而实现了程序的多态性。

>抽象类除了可以包含抽象方法外，其他性质与普通类完全相同。

抽象类自身不能实例化，抽象方法的实现应通过其派生子类来完成。

>接口是更严格的抽象类，其中只能包含public static final成员常量和抽象方法。

>一个子类只能继承一个抽象类，但可以继承多个接口，这实现了程序设计中的多继承关系。

<<Java程序设计实例与操作>>

编辑推荐

《Java程序设计实例与操作》新世纪计算机教育名师课堂，中德著名教育机构精品打造。
边学边练+精彩实例+众多技巧+多媒体教学光盘。

<<Java程序设计实例与操作>>

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>