

<<Java程序设计>>

图书基本信息

书名：<<Java程序设计>>

13位ISBN编号：9787302229759

10位ISBN编号：7302229759

出版时间：2010-7

出版时间：清华大学出版社

作者：幸运韩 等编著

页数：328

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>

前言

Java语言自问世以来，经历了多次的版本升级，从安全机制、语法成分到API函数都有较大的修改。为此我们全面修订了《Java程序设计》，将新版本的内容与Java基础知识一起集成到《Java程序设计》（第二版）中。

为配合主教材的使用，我们编写了本书，旨在帮助读者检查Java语言的学习效果，尽快掌握Java语言。

本书对《Java程序设计》（第二版）一书中12章的全部习题做了完整解答。

对所有的论述题都给出了简单的答案，内容主要摘自《Java程序设计》（第二版）一书。

对所有的编程题目都给出了简单分析，论述了设计思路，并给出了完整的程序代码。

这些题目均在Java 5.0环境下调试通过，程序运行结果以截图的方式提供给读者，以供参考。

此外，本书还精选了9个实验题目（见第13~21章），可供教师配合授课使用。

对每个实验均列出了实验的目的，以此作为检查的目标。

实验后可要求学生完成实验报告，让学生更加深入理解Java语言，确保学生掌握编程技巧。

众所周知，大部分的习题解答不具有唯一性，特别是程序设计题目，读者发挥潜能的余地非常大。

本书中给出的这些解答和参考答案仅供参考，希望能起到抛砖引玉的作用。

因为编者水平的局限性，书中的答案难免存在某些问题，实现的代码也不一定是最优的，读者可以参考本书中的内容和其他参考书中的内容，给出更全面的答案。

至于程序代码，其实现的方式就更加多种多样。

相信读者能在本书的代码基础之上，编写出功能更全面、效率更高的程序。

计算机技术是不断发展、不断完善的技术，Java语言也是如此。

从诞生之日起，它的版本一直在更新中。

就在本书编写过程及读者使用本书期间，相信Java语言又有了新的发展。

读者应及时把握这些新动向，了解最新版本的相关信息，特别是及时更新自己机器上的JDK，以保持设计的代码与新版本的同步。

<<Java程序设计>>

内容概要

本书是和《Java程序设计》(第二版)(幸运伟, 饶一梅, 马素霞编著, 北京, 清华大学出版社出版)一书配套使用的参考书。

书中对《Java程序设计》(第二版)中12章的全部习题给出了完整解答。

其中, 对所有的论述题都给出了简单的答案; 对所有的编程题目都给出了简单分析, 论述了设计思路, 并给出了完整的程序代码。

这些代码均在Java 5.0环境下调试通过, 并将程序运行结果以截图的方式提供给读者, 便于参考。

此外, 本书在第13~21章中, 还精选了9个实验题目, 供教师授课时使用。

通过学习《Java程序设计》(第二版)并配合本书的使用, 能使读者更深入地了解Java语言, 以便熟练掌握它, 并能使用该语言编程完成特定的任务。

本书概念清晰、实用性强, 可供学习Java语言的读者参考使用。

<<Java程序设计>>

书籍目录

1 概述 2 标识符和数据类型 3 表达式和流程控制语句 4 数组、向量和字符串 5 进一步讨论对象和类 6 Java语言中的异常 7 Java的图形用户界面设计 8 Swing组件 9 Java Applet 10 Java数据流 11 线程 12 Java的网络功能 13 实验1 熟悉系统及环境 13.1 实验目的 13.2 实验内容 13.3 参考答案 13.4 程序说明 14 实验2 简单的输入输出处理 14.1 实验目的 14.2 实验内容 14.3 参考答案 14.4 程序说明 15 实验3 类的练习 15.1 实验目的 15.2 实验内容 15.3 参考答案 15.4 程序说明 16 实验4 模拟彩票开奖游戏 16.1 实验目的 16.2 实验内容 16.3 参考答案 16.4 程序说明 17 实验5 模拟CD出租销售店 17.1 实验目的 17.2 实验内容 17.3 参考答案 17.4 程序说明 18 实验6 计算器 18.1 实验目的 18.2 实验内容 18.3 参考答案 18.4 程序说明 19 实验7 用户界面设计 19.1 实验目的 19.2 实验内容 19.3 参考答案 19.4 程序说明 20 实验8 多线程练习 20.1 实验目的 20.2 实验内容 20.3 参考答案 20.4 程序说明 21 实验9 文件读写练习 21.1 实验目的 21.2 实验内容 21.3 参考答案 21.4 程序说明

章节摘录

随着开发系统的不断增大、复杂，面向过程的方法越来越不能满足开发者的要求，面向对象的技术应运而生。

OOP技术使得程序结构简单，相互协作容易，更重要的是程序的重用性大大提高了。

OOP技术把问题看成是相互作用的事物的集合，用属性来描述事物，而把对它的操作定义为方法

。在OOP中，把事物称为对象，把属性称为数据，那么对象就是数据加方法。

OOP中采用了三大技术：封装、继承和多态。

将数据及对数据的操作捆绑在一起成为类，这就是封装技术。

程序员只有一种基本的结构，即类。

将一个已有类中的数据和保留，并加上自己特殊的数据和方法，从而构成一个新类，这是OOP中的继承。

原来的类是父类，新类是子类，父类派生了子类，或说子类继承于父类。

在一个类或多个类中，可以让多个方法使用同一个名字，从而具有多态性。

多态可以保证对不同类型的数据进行等同的操作，名字空间也更加宽松。

使用面向过程编程时，一般地，程序员通过步步细化的过程，将准备完成的任务分解成几个子任务，然后再分解成更小的子任务，直到子任务简单到足以直接编程为止。

这是自顶向下的编程方式。

与此相对应的是自底向上的编程方式，即先编写解决简单任务的过程，并把它们组合成更大更复杂的过程，直到它们的功能满足系统需求为止。

这两种方法都是面向过程编程中经常使用的方法。

在实际实现过程中，程序员往往将两者相结合，灵活利用自顶向下及自底向上的方法。

在实际的开发过程中，面向对象的设计方法和面向过程的设计方法往往相辅相成。

在系统级设计时，经常使用面向对象的设计方法，定义要使用的类及其相关的成员变量和成员方法，定义所需的接口。

而在方法的具体实现时，多采用面向过程的设计方法。

就一个函数或是过程的内部实现来看，面向过程仍然是一种不可缺少的设计方式。

<<Java程序设计>>

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>