

<<Java程序设计与开发>>

图书基本信息

书名：<<Java程序设计与开发>>

13位ISBN编号：9787121116568

10位ISBN编号：7121116561

出版时间：2010-9

出版时间：电子工业出版社

作者：张晓龙 著

页数：248

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程序设计与开发>>

前言

随着互联网的普及和发展,面向Web的程序设计和开发已成为软件编程的新模式,而Java日渐成为互联网编程的一种主流语言之一。

自20世纪90年代早期以来,Java在IT业的应用一直保持增长的趋势。

高等院校的计算机和非计算机专业相继开设了Java程序设计课程。

本书的目的是使学生在理解Java语法特点的基础上,逐步掌握Java语言面向对象的编程要领,同时鼓励学生利用面向对象编程来解决实际问题。

本书以Java语言面向对象的编程理念为出发点,循序渐进地介绍Java语言的基本内容,包括基本语法、数组、字符串、类、抽象类、接口、网络和多线程编程;同时还着重介绍几种Java核心技术,包括JDBC、Servlet、JSP和EJB。

通过融入Java面向对象的编程思想,使读者在学习理论知识的同时,初步掌握面向对象的系统分析、设计与开发能力,为大型软件的设计和开发奠定基础。

本书在讲解Java语言和Java核心技术的同时,辅以大量的实例和结果分析帮助读者快速理解概念的本质,进一步掌握Java语言及其技术的精髓。

本书在实例的安排上,由点到面、由浅入深,并在面向对象的设计思想和技巧上进行深入思考和探讨。

本书第1章主要介绍Java语言的特点和优势,使读者对Java语言有一个概括性的了解。

第2章介绍Java语言基础,主要包括数据类型、变量、表达式、流程控制语句、数组和字符串。

第3章介绍Java类和对象,包括Java类的设计,对象创建和构造方法的执行过程;多态性,类、变量和方法的修饰符(访问和非访问修饰符);内部类,抽象类和接口,包及其作用。

第4章介绍Java的I/O流和异常处理,包括SystemI/O类、Scanner类、File类、字节级输入输出流、字符级输入输出流、对象序列化、异常捕获和处理。

第5章介绍Java图形用户界面和Applet,使用AWT包和Swing包生成图形用户界面,同时使用多种布局管理器设计友好的用户界面。

第6章介绍Java网络编程,包括基于高层URL、基于套接字Socket和基于数据报Datagram3种编程方式,以及使用文件I/O流辅助网络编程。

第7章介绍Java多线程编程,包括线程的创建和启动、线程的优先级、状态切换和多线程的同步处理。

第8章介绍Java数据库连接技术(JDBC),以及利用JDBC编写数据库访问程序的步骤。

第9章介绍Servlet和JSP,包括JavaEEWeb开发概念,JavaBean,Servlet概念及其生命周期,Servlet类的编写、配置和运行,JSP及其编写、配置和运行,JSP的两种设计模式。

第10章介绍RMI和EJB技术,利用RMI机制设计RMI客户端和服务端程序的步骤,介绍了EJB规范、会话类EJB组件的创建和部署,以及编写EJB客户端程序。

本书适合作为计算机及相关专业的本科生、研究生“Java程序设计”课程的教材,也可作为软件开发人员、工程技术人员及其他相关人员作为自学参考书或培训教程。

本书由张晓龙任主编。

其中,第1、3、5章由张晓龙编写;第2、4章由赵红武编写;第6、7章由刘钊编写;第8、9、10章由边小勇编写。

书中的所有实例都在JDK1.6中调试通过。

本书编写过程中,得到了武汉科技大学计算机学院人工智能与机器学习研究室成员的大力支持,在此,谨向他们表示衷心的感谢。

由于时间仓促,编者水平有限,书中难免存在疏漏和不足,恳请读者批评指正,使本书得以改进和完善。

<<Java程序设计与开发>>

内容概要

《Java程序设计与开发》以Java语言面向对象的编程理念为出发点，循序渐进地介绍Java语言的基本内容，包括基本语法、数组、字符串、类和对象、I/O流和异常处理、用户界面类和Applet、网络和多线程编程；同时还着重介绍几种Java核心技术及相应的开发实例，包括JDBC、Servlet、JSP和EJB。通过融入Java面向对象的编程思想，使读者在学习理论知识的同时，初步掌握面向对象的系统分析、设计与开发能力，为大型软件的设计和开发奠定基础。

《Java程序设计与开发》配有PPT、源代码、习题答案等教学资源。

《Java程序设计与开发》可作为高等学校计算机及相关专业“Java程序设计”课程的教材，也可作为软件开发人员、工程技术人员及其他相关人员的自学参考书或培训教程。

书籍目录

第1章 Java语言概述 (1) 1.1 Java语言及其产生 (1) 1.2 Java的特点 (1) 1.3 Java的体系结构 (3) 1.3.1 Java的体系结构 (3) 1.3.2 Java虚拟机 (3) 1.3.3 垃圾回收机制 (3) 1.3.4 代码安全检测 (4) 1.4 Java的3种版本 (4) 1.5 Java开发环境概述 (4) 1.5.1 JDK的安装、配置和使用 (4) 1.5.2 JDK工具箱 (6) 1.5.3 Eclipse开发环境的安装、配置和使用 (6) 1.5.4 Java程序的编写 (7) 1.6 本章小结 (9) 习题1 (9) 第2章 Java语言基础 (11) 2.1 Java语言要素 (11) 2.1.1 标识符 (11) 2.1.2 关键字 (11) 2.1.3 分隔符 (12) 2.1.4 注释 (12) 2.2 基本数据类型 (13) 2.2.1 简单类型 (13) 2.2.2 引用类型 (14) 2.2.3 常量 (14) 2.2.4 变量 (15) 2.3 运算符和表达式 (16) 2.3.1 算术运算符 (16) 2.3.2 关系运算符 (17) 2.3.3 逻辑运算符 (17) 2.3.4 位运算符 (18) 2.3.5 赋值运算符 (19) 2.3.6 其他运算符 (20) 2.3.7 运算符的优先级与结合性 (21) 2.3.8 表达式与语句 (21) 2.4 控制语句 (22) 2.4.1 选择结构控制语句 (23) 2.4.2 循环结构控制语句 (25) 2.4.3 转移语句 (27) 2.5 数组 (29) 2.5.1 一维数组 (29) 2.5.2 二维数组 (31) 2.5.3 多维数组 (32) 2.6 字符串 (32) 2.6.1 字符串类 (33) 2.6.2 常用方法 (33) 2.7 综合应用实例 (34) 2.8 本章小结 (37) 习题2 (37) 第3章 Java类和对象 (39) 3.1 面向对象基础 (39) 3.1.1 类 (39) 3.1.2 对象 (40) 3.1.3 消息 (40) 3.2 继承 (40) 3.2.1 创建子类 (40) 3.2.2 成员变量和方法 (41) 3.2.3 对象的构造和初始化过程 (44) 3.2.4 上转型对象 (45) 3.2.5 关键字this和super (47) 3.2.6 对象的撤销和清理 (49) 3.3 多态 (50) 3.3.1 基本概念 (50) 3.3.2 方法重载 (50) 3.3.3 方法覆盖 (51) 3.3.4 变量覆盖 (53) 3.4 修饰符 (53) 3.4.1 访问修饰符 (53) 3.4.2 非访问修饰符 (57) 3.5 内部类和匿名内部类 (59) 3.5.1 内部类 (59) 3.5.2 匿名内部类 (61) 3.6 接口 (63) 3.6.1 接口的定义 (63) 3.6.2 接口与多重继承 (63) 3.6.3 接口的实现 (64) 3.6.4 接口中的变量和方法 (64) 3.7 包 (66) 3.7.1 包的作用 (66) 3.7.2 Java中定义的包 (66) 3.7.3 自定义包的创建和使用 (66) 3.7.4 编译和生成包 (67) 3.8 综合应用实例 (68) 3.9 本章小结 (70) 习题3 (71) 第4章 Java的I/O流和异常处理 (73) 4.1 I/O流概述 (73) 4.2 System I/O类和Scanner类 (75) 4.2.1 System I/O类 (75) 4.2.2 Scanner类 (76) 4.3 文件类 (76) 4.3.1 使用File类访问文件系统 (76) 4.3.2 随机文件读写 (78) 4.4 字节级输入输出流 (79) 4.4.1 数据输入输出流 (79) 4.4.2 文件输入输出流 (80) 4.4.3 缓冲输入输出流 (81) 4.4.4 管道输入输出流 (81) 4.4.5 格式化输出流 (81) 4.5 字符级输入输出类 (82) 4.5.1 字符输入输出类 (82) 4.5.2 缓冲读写类 (83) 4.5.3 文件读写类 (86) 4.5.4 格式化写类 (88) 4.6 对象序列化 (89) 4.6.1 对象输入流 (89) 4.6.2 对象输出流 (90) 4.7 异常处理 (91) 4.7.1 Java中定义的异常 (91) 4.7.2 异常类的层次结构 (92) 4.7.3 捕获异常 (92) 4.7.4 异常处理的嵌套 (96) 4.7.5 自定义的异常和使用 (97) 4.8 文件流综合应用实例 (98) 4.9 本章小结 (106) 习题4 (106) 第5章 Java用户界面类和Applet (108) 5.1 AWT简介 (108) 5.2 AWT容器类 (108) 5.3 基本控件类 (110) 5.4 图形类 (112) 5.5 事件类型及其处理 (112) 5.5.1 事件类 (113) 5.5.2 事件监听器 (113) 5.5.3 事件处理流程 (114) 5.6 布局管理器 (115) 5.6.1 FlowLayout (115) 5.6.2 BorderLayout (116) 5.6.3 GridLayout (116) 5.6.4 CardLayout (116) 5.6.5 GridBagLayout (117) 5.7 Swing简介 (117) 5.7.1 Swing组件的特点 (118) 5.7.2 Swing包中的类 (118) 5.7.3 Swing包中的控件类 (118) 5.8 Java Applet (122) 5.8.1 Applet简介 (122) 5.8.2 Applet程序的编写 (124) 5.8.3 Applet程序的编译和执行 (125) 5.8.4 Applet的绘图和控制 (126) 5.9 GUI设计综合应用实例 (128) 5.10 本章小结 (130) 习题5 (130) 第6章 Java网络编程 (132) 6.1 Java网络基础 (132) 6.1.1 IP地址 (132) 6.1.2 端口 (133) 6.1.3 套接字 (134) 6.1.4 数据报 (134) 6.1.5 传输协议 (135) 6.2 基于URL的高层次Java网络编程 (135) 6.2.1 从URL中读取内容 (135) 6.2.2 建立URL连接并读取内容 (136) 6.3 基于Socket的网络编程 (137) 6.3.1 Socket通信过程 (138) 6.3.2 创建客户端Socket (139) 6.3.3 创建服务器端Socket (139) 6.3.4 打开输入输出流 (139) 6.3.5 关闭流和Socket (141) 6.4 基于Datagram的网络编程 (141) 6.4.1 数据报和套接字 (142) 6.4.2 创建客户端Datagram (142) 6.4.3 创建服务器端Datagram (142) 6.4.4 打开输入输出流 (143) 6.4.5 关闭流和Datagram (143) 6.5 网络编程与应用实例 (144) 6.6 本章小结 (150) 习题6 (150) 第7章 Java多线程编程 (152) 7.1 线程基础 (152) 7.1.1 线程概述 (152) 7.1.2 线程的状态 (152) 7.1.3 线程的优先级 (153) 7.2 线程的创建 (153) 7.2.1 通过继承Thread类创建 (153) 7.2.2 通过实现Runnable接口创建 (155) 7.3 多线程的创建 (156) 7.4 线程的调度 (159) 7.4.1 线程优先级的获取与设置 (159) 7.4.2 线程睡眠 (159) 7.4.3 线程让步 (159) 7.4.4 线程等待 (160) 7.4.5 线程唤醒 (160) 7.5 多线程的

同步处理 (161) 7.5.1 多线程的互斥 (161) 7.5.2 synchronized方法 (161) 7.6 多线程综合应用实例 (164)
7.7 本章小结 (168) 习题7 (168) 第8章 数据库编程 (170) 8.1 数据库基础 (170) 8.1.1 关系数据库
(170) 8.1.2 SQL基本内容 (171) 8.1.3 ODBC技术 (173) 8.2 JDBC数据库连接技术 (174) 8.2.1 JDBC概
述 (174) 8.2.2 JDBC的实现及其驱动程序 (174) 8.2.3 JDBC的常用类和接口 (175) 8.3 Java访问数据库
的基本步骤 (176) 8.3.1 加载驱动程序类 (176) 8.3.2 建立JDBC连接 (177) 8.3.3 执行SQL语句 (179)
8.4 JDBC高级特性 (180) 8.4.1 预备语句 (180) 8.4.2 可滚动和可更新的结果集 (180) 8.4.3 元数据
(182) 8.4.4 事务 (183) 8.5 创建数据库 (184) 8.6 JDBC综合应用举例 (185) 8.7 本章小结 (187) 习
题8 (188) 第9章 Servlet与JSP (189) 9.1 Java EE Web开发技术 (189) 9.2 Servlet基础 (190) 9.2.1 Servlet
定义 (190) 9.2.2 Servlet工作方式 (191) 9.2.3 Servlet API (192) 9.2.4 Servlet异常 (193) 9.3 配置Servlet工
作环境 (193) 9.3.1 Servlet开发环境 (193) 9.3.2 Servlet类的配置 (194) 9.3.3 编写简单的Servlet (195)
9.4 JavaBean基础 (196) 9.4.1 JavaBean基本概念 (196) 9.4.2 JavaBean的属性、方法和事件 (197) 9.4.3
一个简单的JavaBean (197) 9.5 JSP概述 (198) 9.5.1 JSP基本概念 (198) 9.5.2 JSP技术的特点 (199) 9.5.3
JSP的工作方式 (199) 9.5.4 JSP和Servlet的区别 (200) 9.6 编写简单的JSP (200) 9.7 在JSP中访问数据库
(200) 9.8 JSP的设计模式 (202) 9.8.1 JavaBean与JSP的结合 (202) 9.8.2 Servlet与JSP的结合 (203) 9.9
JSP的语法要素 (203) 9.9.1 JSP脚本元素 (203) 9.9.2 JSP指令和动作元素 (205) 9.9.3 JSP内置对象 (206)
9.10 Web综合应用实例 (209) 9.11 本章小结 (222) 习题9 (223) 第10章 RMI和EJB (224) 10.1 RMI
基础 (224) 10.1.1 RMI概述 (224) 10.1.2 RMI的目标和体系结构 (224) 10.1.3 RMI的工作机制 (225)
10.1.4 用RMI设计应用系统 (226) 10.2 RMI API (227) 10.3 Java RMI与其他分布式技术 (228) 10.4
RMI部署问题 (229) 10.5 编写一个RMI应用系统 (229) 10.6 EJB技术 (233) 10.6.1 EJB基本概念 (233)
10.6.2 EJB组件 (234) 10.6.3 EJB容器 (235) 10.6.4 开发和部署EJB (236) 10.6.5 EJB和其他分布式技术
(237) 10.6.6 基于EJB的Web应用程序实例 (238) 10.7 本章小结 (244) 习题10 (245) 附录A Java运算
符及其优先级 (246) 附录B Java类路径和类加载器 (247) 参考文献 (249)

<<Java程序设计与开发>>

章节摘录

JVM (Java虚拟机) 是Java Virtual Machine的缩写。它是一个虚构出来的计算机, 是通过在实际的计算机上模拟各种计算机功能来实现的。Java虚拟机有自己完善的硬件架构, 例如处理器、堆栈、寄存器等, 还具有相应的指令系统。Java语言的一个非常重要的特点就是与平台无关性, 而使用Java虚拟机是实现这一特点的关键。一般的高级语言要在不同的平台上运行时, 至少需要编译成不同的目标代码。而引入Java虚拟机后, 它屏蔽了与具体平台相关的信息, 使得Java编译程序只需生成能在Java虚拟机上运行的目标代码(字节码), 然后, Java虚拟机把字节码解释成具体平台上的机器指令执行。因此, Java程序无须重新编译就可以跨平台运行。

JVM在执行应用程序时具有以下3个典型特点。

多线程: Java虚拟机采用多线程方式运行, 各线程独立执行Java代码, 并能够协调一致地处理共享数据。

动态性: Java虚拟机在内存管理等功能方面全部采用动态方式, 如动态分配数组的存储空间、动态创建对象、动态连接数据库等, 这些动态特性使Java程序适合在网络上运行。

异常处理机制: Java虚拟机提供可靠的异常处理机制。

1.3.3 垃圾回收机制 在Java语言中, 所有属性和方法都被封装在类中。

当要创建类的对象时, 这种动态的对象被分配到内存堆中。

为了实现内存资源的合理使用, Java有一个系统级的线程, 定期跟踪内存的使用情况。

此线程在虚拟机的空闲周期检测内存, 对内存中已经死亡或者长时间没有使用的对象进行清除和回收。

当然, Java虚拟机在一定条件下才执行内存收集。

例如可用内存较少、CPU占用率较低, 或者其他的策略。

因此, 编程人员无须考虑复杂的内存管理。

.....

<<Java程序设计与开发>>

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>