

SAMS

在文川网搜索古籍书城 获取更多电子书
入驻商家
docsriver文川网
古籍书城

Java 2

Primer Plus

中文版



[美] Steven Haines, Stephen Potts 著
汪青青 朱剑平等 译

 人民邮电出版社
POSTS & TELECOM PRESS

Java 2 Primer Plus 中文版

[美] Steven Haines, Stephen Potts 著

汪青青 朱剑平 等 译

人民邮电出版社

图书在版编目 (CIP) 数据

Java 2 Primer Plus 中文版 / (美)海恩斯 (Haines,S.), (美)波茨 (Potts,S.) 著; 汪青青等译. —北京: 人民邮电出版社, 2003.11

ISBN 7-115-11794-2

I. J... II. ①海...②波...③汪... III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 092435 号

版权声明

Steven Haines, Stephen Potts: Java 2 Primer Plus (ISBN: 0-672-32415-6)

Copyright © 2003 by Sams Publishing

Authorized translation from the English language edition published by the Sams Publishing.

All rights reserved.

本书中文简体字版由美国 Sams 公司授权人民邮电出版社翻译出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

Java 2 Primer Plus 中文版

- ◆ 著 [美] Steven Haines, Stephen Potts
译 汪青青 朱剑平 等
责任编辑 李 岚
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132705
北京汉魂图文设计有限公司制作
北京隆昌伟业印刷有限公司 印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 26.5
字数: 840 千字 2003 年 11 月第 1 版
印数: 1-4 000 册 2003 年 11 月北京第 1 次印刷

著作权合同登记 图字: 01 - 2002 - 3670 号

ISBN 7-115-11794-2/TP · 3715

定价: 48.00 元 (附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

docsriver 文川网
入驻商家 古籍书城

在文川网搜索古籍书城 获取更多电子书

内 容 提 要

本书全面而深入地讲解 Java 编程知识，并揭示 Java 各种特性的设计根源和目标，帮助读者学习并理解 Java 的本质和精髓。全书分为五大部分，共 26 章和 3 个附录，内容包括：Java 基础、面向对象编程、图形用户界面、高级主题和 Java Web 技术。每章最后配有一组复习题和练习题，帮助读者巩固本章所学知识。附录部分提供了习题答案，并提供了 Borland JBuilder 以及 Sun One Studio 4 的使用说明。

本书适合初学计算机编程的读者或初学 Java 的程序员作为全面学习 Java 的教材。希望对 Java 获得更深理解的 Java 程序员也将在本书中受益匪浅。

译者序

Java 以其优秀的跨平台能力而风靡全球。本书全面而深刻地揭示了 Java 简单、面向对象、解释执行、安全、跨平台、高性能、多线程等诸多迷人的特性，并深入讨论了如何使用 AWT 和 Swing 开发用户界面，如何使用 Java 技术开发各种组件、数据库应用及 Web 站点。书中的示例丰富而详尽，讲解清晰而透彻；尤其值得一提的是，本书试图从 Java 创始人的角度来揭示 Java 各种特性的设计根源和目标，从而帮助读者理解 Java 的本质和精髓。本书从最基本的编程知识入手，因而适合各类读者，即便是那些对计算机编程一无所知的读者也能跟随本书轻松入门，并继而逐步领会 Java 的全部精华。

参加本书翻译的人员有：汪青青、朱剑平、刘颖、吴东升、杨战伟、谢文亮、朱志博、江东海、汪佳等。汪青青负责全书的统稿。负责本书版式工作的是杨晓桃。在此向北京静影文化工作室的全体人员表示感谢！

感谢人民邮电出版社外版图书编辑部同仁的不懈努力和大力支持！他们严谨的工作态度也令我们受益匪浅。

限于译者水平加之时间仓促，必有纰漏或错误之处，恳请读者和专家不吝指正。希望这本书能引领您步入 Java 编程的神圣殿堂！

译者

2003年6月

关于作者

Steven Haines 在过去的 8 年里一直从事企业软件行业工作，并且自 1997 年起便专注于 Java。在过去的几年里，Steven Haines 曾先后主持过 B2B 电子商务、高速 Internet 市场、应用程序监控和诊断、健壮的客户端和服务端映像布局和管理等领域的关键体系结构设计。目前，他是 Quest Software 公司的 J2EE 系统工程师，负责为调谐和监控 Enterprise Java 应用程序和应用程序服务器定义专家规则。

Steven Haines 是 Que 出版社 *Java 2 from Scratch* 一书的作者，并在 InformIT.com 上发表了大量有关 Java Swing 和 Enterprise Java 的文章。他还与他人合作撰写了 *Java Web Services Unleashed*、*C++ Unleashed*、*Sams Teach Yourself C++ in 21 Days* 和 *Sams Teach Yourself Java in 21 Days* 等著作。他还是 Pearson Education 的技术编辑，负责 Java、Enterprise Java、网络通信、C++ 和视频及游戏编程的编辑工作。Steven 曾在 Learning Tree 大学 (LTU) 教授各种 Java 编程，从基本的认证培训到数据库、Web 开发和 Enterprise JavaBean。

Stephen Potts 是乔治亚州亚特兰大市的一名独立顾问、并兼做 Java 教员。Stephen 在 1982 年获得了 Georgia Tech (乔治亚工学院) 的计算机科学学位。在 20 年的职业生涯中，他曾教授过多门学科，其中制造业是他的最爱。Stephen 曾担任 Home Depot、Disney 和 IBM 等公司的顾问。他撰写的图书包括 *Java Unleashed* 和 *Java 1.2 How-To*。可以通过电子邮件 stevepotts@mindspring.com 与他联系。

天子技术编辑

John Purdum 是印第安那州印第安纳波利斯市 Roche Diagnostics 公司的软件工程师，他目前正在开发针对企业信息仓库的单机及基于 Web 的 Java 应用程序，专门负责可重用组件的开发。在担任这份工作之前，他是 Bank One 公司 Visual Basic 和 C++ 开发方面的软件工程顾问。用 Java 语言编程是他职业生涯最光辉的一个方面。

致 谢

首先，我想感谢我在 Sams 出版社的所有朋友，感谢他们帮助我完成本书：Michelle Newcomb、Todd Green、Mark Renfrow、Andrew Beaster、Chip Gardner 和 Heather McNeill。特别感谢 Michelle Newcomb，有你的帮助乃是我的荣幸，在过去的几年里，你成为我伟大的朋友！感谢我的家人对我的支持：Elizabeth Haines、Heru Susilo、Cecilia Chandra、Herindarno Heru（Wei Wei，薇薇）、Hernanto Heru（Yong Yong，勇勇），以及我最亲爱的伯父 Bing Oei。感谢我的朋友默默地支持我的计划：Jeff 和 Maria Thornton、Matt Torres、Alex Blank、Henry 和 Yolanda Jahja。非常感谢 Quest Software 公司技术人员的支持：Guy Harrison、Eyal Aranoff 和 Pat Stephenson。感谢 8 年来曾与我共事并保持联系的计算机界朋友——Canon 公司的 Jessie Labayen、Mark An、Jay Tadbiri、Raemonde Olson、Kiyoshi Oka、Julie Lin；Engage 公司的 Jyoti Mody、Gary McKenzie、Siobhan Cooper；Wonderware 公司的 Robert Asis、Trevor Nguyen、Keyee Hsu、Jay Cook、Rashesh Mody；Rainbow 公司的 Anne Barrette；IPNet 公司的 Danny Lim、Ryan Saldara、Sean Slavin、Gary Kostalnick、Steve Chuang；以及 BigBallot 公司的 Michael Foo。感谢 LTU 的朋友：Mara Scott、Sandy Ock、Brad O’Hearne 和 Brian Maslowe。最后，我还想感谢您，我的读者——在市场上有很多 Java 方面的图书，我很荣幸您选择了这本书——我不会让您失望的！

——Steven Haines

我想感谢本书所有的编辑，感谢他们作出的奉献。Michelle Newcomb 让我投入这个项目，Todd Green 帮助我坚持我的计划。Mark Renfrow 在本书的细节内容上给予我很大的帮助。我还想感谢技术审稿人 John Purdum 和 Marc Goldford，感谢他们的努力。

——Stephen Potts

前言

Java 编程，与所有计算机编程一样，需要刻苦钻研的精神和合适的学习工具。作为一名 Java 程序员，我曾花费了数年的时间阅读了大量的 Java 图书，这些图书涉及到 Java 的方方面面，并分析了 Sun 对 Java 的实现来深入了解它的设计原理。我所读过的 Java 编程书籍在细节和方法上差别很大，但都遗漏了某些非常重要的东西。

几年前，当我深入学习 Windows 编程时，我研读了 Don Box 的 *Essential COM*，这本书改变了我对编程书的标准。COM 是一个难以理解的主题，更别说精通 COM，但 Don 的方法与市场上其他同类图书的方法大不相同。他从基础知识入手，并对 COM 的功能逐一演绎，因此他的读者能够真正理解每一个设计决定的原因。到读完第一章时，我对 COM 的理解已经达到完全不同的境界：COM 不再那么神秘，一切都突然变得富有趣味。在那以后，当学习 COM 更高级的特性时，只不过是正在一个已打好的坚实地基上再添砖加瓦罢了。

我从未看到过有 Java 书采用这种方法，这正是《Java 2 Primer Plus 中文版》一书与其他图书的不同之处。不是简单地教读者如何编写 Java 程序，而是倾注全部的笔墨来告诉读者这门语言为什么是这样的。该书还探索了 Java 创始者在创建该语言时所作出的底层设计决策。因此，本书的目标是帮助读者打下 Java 编程语言的坚实基础，在此基础上通读全书，就像新技术的发展一样，读者可以在他坚实的基础上添枝加叶。

本书的读者对象

《Java 2 Primer Plus 中文版》适合以下各类读者：

- 初学计算机编程的读者；
- 初学 Java 的程序员；
- 希望对 Java 获得更深理解的 Java 程序员。

本书从 Java 基础知识开始，并假设读者对计算机编程一无所知。本书为编程新手和经验丰富的专业人士都提供了足够深度的内容。如果您已精通另一门编程语言，那么在学习本书的前几章时您可以将注意力集中在几个应用程序示例上，以便熟悉 Java 编程语言的关键概念。如果您只是普通的计算机编程新手，本书将指导您通晓基本的概念并解释这些概念的根源。

本书的编写工作不仅依靠我作为一名专业 Java 程序员的经验，还从课堂中借用了久经试验的教学经验。我曾教过各种程度的学生，从计算机编程新手到 Visual Basic 专家，从会计师到经验丰富的 Java 程序员，并设计了一套 Java 编程方法，这些方法用于特定的情况。我的目标是将这些经验发布出来。

阅读本书的预备知识

您不需要任何编程背景，也将会发现这本书是极有价值的学习助手。

本书的组织结构

本书分为 5 个部分：

- 第一部分：Java 基础。
- 第二部分：面向对象编程。
- 第三部分：图形用户界面。
- 第四部分：高级主题。
- 第五部分：Web 技术。

第一部分“Java 基础”覆盖了适合 Java 编程的计算机编程基础知识。该部分讲述了 Java 的关键字、数据类型、变量、运算符、流控制和方法。在通读“Java 基础”之后，您将对本书后面构建 Java 程序时所需用到的主要工具有所了解。

第二部分“面向对象编程”介绍了迄今为止计算机编程领域中最流行、最有效的编程范例——面向对象编程。该部分讲述了 Java 类、封装、继承和多态。此外，该部分还包括了一些更高级的 Java 核心功能，包括接口、异常处理、Java 集合类和输入与输出。

第三部分“图形用户界面”指导您如何使用标准抽象窗口工具为应用程序和小程序（applet）开发图形用户界面以及如何使用 Swing 来开发高级用户界面。该部分带领您全面地探索驱动 Java 整个事件处理策略的事件委派模型。

第四部分“高级主题”汇总了被我称为 Java 编程语言核心组件的全部高级主题。在学完该部分之后，读者应该能够理解应用于 Web、企业计算或移动计算的任何 Java 应用程序。讲述的主题包括多线程编程、Java 自省和反射编程接口、Java 数据库连接 API（JDBC）、使用原始协议的网络编程、Java 远程方法调用（RMI）。该部分以对最佳实践和代码优化技术的讨论收尾。

第五部分“Web 技术”讨论了在服务器端 Web 编程中采用 Java 语言编写的应用程序。该部分从最基本的 servlet 开始讨论，辅之以最新、易于使用的 JavaServer 页面（JSP），此外还讲述了诸如如何构建自定义标记库来强化 Web 编程和处理 XML 之类的高级技术。最后，该部分还包括了对何谓正确的 Web 应用程序设计的讨论，并讨论了应用于 Web 设计中的、久经考验的模型—视图—控制器（MVC）设计模式。由于 Java Web 编程非常灵活，因而很容易滥用 Java 技术而导致性能不佳；该部分能够满足您的需要，可以确保您的 Web 应用程序是可扩展的并且经过优化配置。

本书中使用的约定

本书中的程序清单均收录在随书光盘中，在阅读本书时，请读者参看光盘中的相应内容，一并学习。

每章配有一组复习题和练习题。目的是巩固 Java 编程语言的关键概念并鼓励您通过将所学知识应用到实际项目中来深化学习。学习编程语言是项主动性练习，无论想精通哪门编程语言，都不能仅仅只阅读相关书籍，而必须将所学的知识应用到实际的示例中。我鼓励您做这些练习题；做完之后，您的能力将会大大提高！

本书中的其他内容包括：

注释

注释部分是对正在讨论的主题的注解和旁白，或是对特定概念的完整解释。

提示

提示部分给出极好的捷径和提示，以便更有效地进行 Java 2 编程。

警告

警告部分警告您注意避免常见的错误。

目 录

第一部分 Java 基础

第 1 章 Java 简介	3
1.1 Java 是什么	4
1.1.1 历史	4
1.1.2 为什么使用 Java	4
1.1.3 Java 的表征性术语	5
1.2 建立 Java 编程环境	6
1.2.1 Microsoft Windows	6
1.2.2 Linux	6
1.3 编写第一个 Java 程序	7
1.3.1 编译第一个 Java 程序	7
1.3.2 运行第一个 Java 程序	9
1.3.3 理解第一个 Java 程序	9
1.4 小结	11
第 2 章 关键字、数据类型和变量	12
2.1 编程新手	13
2.2 关键字	13
2.3 数据类型	13
2.3.1 人/机通信问题	14
2.3.2 基元数据类型	15
2.4 变量	17
2.5 字面值	19
2.6 数值类型之间的转换	19
2.7 数据类型之间的强制转换	21
2.8 常量	21
2.9 小结	22
2.10 复习题	22
第 3 章 运算符	23
3.1 算术运算符	24
3.1.1 加法	24
3.1.2 减法	24
3.1.3 乘法	24
3.1.4 除法	25
3.1.5 模数	25
3.2 递增和递减运算符	26
3.3 关系运算符	26
3.4 位运算符	27
3.4.1 与	28

3.4.2 或	29
3.4.3 异或	29
3.4.4 非运算符	29
3.4.5 实际的应用程序	29
3.5 逻辑运算符	30
3.5.1 与	30
3.5.2 或	30
3.5.3 短路运算符	31
3.6 移位运算符	31
3.6.1 左移运算符	32
3.6.2 右移	32
3.6.3 右移(补零)	33
3.7 运算符的优先级	33
3.8 小结	35
3.9 复习题	35
第4章 流控制	36
4.1 条件语句	37
4.1.1 if-then-else 语句	37
4.1.2 switch 语句	40
4.2 循环结构	42
4.2.1 while 语句	42
4.2.2 do 语句	42
4.2.3 for 语句	43
4.2.4 跳出循环	45
4.2.5 在循环中继续执行	45
4.2.6 标签	46
4.3 小结	47
4.4 复习题	47
4.5 练习题	47
第5章 方法	48
5.1 分而治之	49
5.2 方法的格式	49
5.3 调用方法	50
5.4 复杂的类方法	51
5.5 变量作用域	52
5.6 递归	52
5.7 方法的重载	53
5.8 Java 类库	54
5.9 小结	56
5.10 复习题	56
5.11 练习题	56

第二部分 面向对象编程

第 6 章 类	59
6.1 面向对象程序设计	62
6.1.1 封装	62
6.1.2 信息隐藏	62
6.2 类	62
6.2.1 CarObject 类	62
6.2.2 组件属性	63
6.2.3 状态属性	63
6.2.4 行为（方法）	63
6.2.5 Get/Set 方法	63
6.2.6 构造函数	64
6.2.7 类范围	65
6.2.8 控制对成员的访问	65
6.2.9 this 变量	65
6.2.10 常量	65
6.2.11 组合	66
6.2.12 垃圾收集	66
6.2.13 终结器	66
6.2.14 静态类成员	67
6.3 内部类	67
6.3.1 范围	67
6.3.2 访问修饰符	68
6.3.3 静态内部类	68
6.3.4 匿名内部类	68
6.4 理解对象引用	68
6.5 小结	69
6.6 复习题	69
6.7 练习题	69
第 7 章 继承	70
7.1 问题的由来	71
7.2 解决方案：接口	73
7.3 何时使用继承	75
7.4 继承关系的准则	75
7.5 树形图	76
7.6 语法	76
7.6.1 final 类	76
7.6.2 抽象类	76
7.6.3 访问修饰符	77
7.7 类的终结器	77
7.8 重载方法	78
7.9 动态绑定和多态	78

7.10	小结	78
7.11	复习题	79
7.12	练习题	79
第 8 章	接口	80
8.1	问题的由来	81
8.2	多重继承	82
8.3	接口语法	83
8.4	实现接口	83
8.5	使用接口实现多重继承	84
8.6	使用接口为类添加外部性能	84
8.7	接口与抽象类的对比	84
8.8	小结	84
8.9	复习题	85
8.10	练习题	85
第 9 章	异常处理	86
9.1	问题的由来	87
9.2	异常的定义	90
9.2.1	try 块	90
9.2.2	catch 块	90
9.2.3	finally 块	90
9.3	异常的语法	91
9.4	异常类	92
9.5	引发异常	94
9.5.1	声明异常列表	94
9.5.2	创建异常实例	94
9.5.3	引发异常	94
9.6	重新引发异常	95
9.7	通过方法传递异常	95
9.8	创建自定义异常	95
9.9	小结	96
9.10	复习题	97
9.11	练习题	97
第 10 章	Java 类	98
10.1	JavaDoc 简介	99
10.2	使用 Java 的包装类	100
10.2.1	Boolean 类	101
10.2.2	Byte 类	101
10.2.3	Character 类	102
10.2.4	Double 类	103
10.2.5	Float 类	104
10.2.6	Integer 类	105
10.2.7	Long 类	106

10.2.8 Short 类.....	107
10.3 使用 String 类.....	108
10.3.1 String 类.....	109
10.3.2 StringBuffer 类.....	112
10.3.3 StringTokenizer 类.....	113
10.4 小结.....	114
10.5 复习题.....	115
10.6 练习题.....	115
第 11 章 集合类.....	116
11.1 集合类解决的问题.....	117
11.2 数据结构简介.....	117
11.2.1 数组数据结构.....	117
11.2.2 链表数据结构.....	118
11.2.3 堆栈数据结构.....	120
11.2.4 队列数据结构.....	121
11.2.5 哈希表数据结构.....	121
11.2.6 树数据结构.....	123
11.3 集合概念.....	124
11.3.1 列表.....	125
11.3.2 Set 接口.....	125
11.3.3 Map 接口.....	125
11.3.4 Iterator.....	125
11.4 集合类.....	125
11.4.1 ArrayList.....	126
11.4.2 向量.....	127
11.4.3 LinkedList.....	128
11.4.4 Stack 类.....	129
11.4.5 HashSet.....	129
11.4.6 TreeSet.....	130
11.4.7 HashMap.....	131
11.4.8 TreeMap.....	132
11.5 小结.....	133
11.6 复习题.....	133
第 12 章 使用 Java 的输入和输出类.....	135
12.1 问题的由来.....	136
12.2 作为输入和输出数据源的流.....	137
12.3 面向字节的流类.....	137
12.3.1 预定义的流对象.....	138
12.3.2 过滤流.....	138
12.3.3 读写文件.....	139
12.3.4 其他字节 I/O 类.....	141
12.4 字符流.....	141
12.4.1 使用 Reader/Writer 执行文件 I/O.....	142

12.4.2	字节和字符 I/O 类	143
12.4.3	执行其他字符 I/O 的 Reader 和 Writer 类	143
12.4.4	StreamTokenizer	143
12.5	链接流以便使用	144
12.6	小结	144
12.7	复习题	144
12.8	练习题	144

第三部分 图形用户界面

第 13 章	抽象窗口工具包	147
13.1	容器	148
13.1.1	Applet	149
13.1.2	运行小程序	149
13.1.3	Frame	150
13.1.4	Panel	154
13.1.5	ScrollPane	154
13.2	对话框编程	156
13.3	向 GUI 添加 AWT 组件	157
13.3.1	TextField	157
13.3.2	添加按钮	158
13.3.3	TextArea	160
13.3.4	CheckBox	161
13.3.5	CheckboxGroup	162
13.3.6	选择控件编程	163
13.3.7	列表控件编程	163
13.3.8	MenuBar	165
13.3.9	MenuItem	166
13.4	布局管理器	166
13.4.1	BorderLayout 管理器	167
13.4.2	FlowLayout	168
13.4.3	GridBagLayout	168
13.5	小结	169
13.6	复习题	170
13.7	练习题	170
第 14 章	事件委派模型	171
14.1	理解事件	172
14.2	Java 事件模型编程	173
14.3	其他事件监听器接口的使用	175
14.3.1	ActionListener 接口	176
14.3.2	AdjustmentListener 接口	177
14.3.3	ItemListener 接口	178
14.3.4	TextListener 接口	179
14.3.5	KeyListener 和 KeyAdapter 接口	180

docsriver 文川网
入驻商家 古籍书城

在文川网搜索古籍书城 获取更多电子书

14.3.6 WindowListener 和 WindowAdapter 接口.....	181
14.3.7 ComponentListener 和 ComponentAdapter 接口.....	183
14.3.8 MouseListener 和 MouseAdapter 接口.....	183
14.3.9 MouseMotionListener 和 MouseMotionAdapter 接口.....	184
14.4 小结.....	185
14.5 复习题.....	185
14.6 练习题.....	186
第 15 章 JavaBean	187
15.1 理解 JavaBean.....	188
15.2 开发简单的 JavaBean.....	189
15.3 在 Forte 中部署.....	193
15.4 属性、方法和事件.....	195
15.4.1 创建新方法.....	195
15.4.2 事件.....	195
15.4.3 PropertyChangeEvent.....	196
15.5 小结.....	198
15.6 复习题.....	199
15.7 练习题.....	199
第 16 章 Swing 库	200
16.1 理解 Swing.....	201
16.2 Swing 应用程序结构.....	202
16.2.1 窗口管理.....	203
16.2.2 JFrame.....	204
16.2.3 JComponent.....	204
16.2.4 JPanel.....	204
16.2.5 使用 JScrollPane 控件.....	205
16.3 Dialog.....	206
16.4 使用 Swing 组件.....	207
16.4.1 JTextField 组件.....	207
16.4.2 添加按钮.....	208
16.4.3 JTextArea.....	209
16.4.4 JCheckBox.....	211
16.4.5 JRadioButton.....	212
16.4.6 JList.....	212
16.5 小结.....	213
16.6 复习题.....	213
16.7 练习题.....	213
第 17 章 高级 Swing	215
17.1 JTable 编程.....	216
17.1.1 添加标题、滚动条并格式化.....	218
17.1.2 JTree 编程.....	219
17.2 使用 JOptionPane.....	222

17.3	小结	223
17.4	复习题	223
17.5	答案	223
17.6	练习题	224

第四部分 高级主题

第 18 章	多线程编程	227
18.1	为什么需要多线程	228
18.2	使用 Thread 类控制线程	228
18.2.1	创建多线程应用程序	229
18.2.2	设置线程名称	230
18.3	使用 Runnable 接口创建线程	231
18.3.1	中断线程	232
18.3.2	结束线程	233
18.4	协调线程对变量的访问	234
18.5	线程之间的协调	236
18.6	在线程间发送数据	237
18.7	小结	239
18.8	复习题	239
18.9	练习题	239
第 19 章	使用 Java 数据库连接 (JDBC) 访问数据库	240
19.1	了解数据库的历史	241
19.2	理解为何需要 JDBC	242
19.3	Java 数据库连接编程	243
19.4	查询数据库	245
19.5	使用 JDBC 创建表	247
19.6	填充数据库	248
19.7	使用 JDBC 和 GUI 来改变数据	249
19.8	小结	251
19.9	复习题	251
19.10	练习题	251
第 20 章	网络编程	252
20.1	TCP/IP 的历史	253
20.2	TCP/IP 内幕	253
20.3	应用层	254
20.4	传输层	254
20.5	网络层	255
20.6	网络接口层	255
20.7	简单的套接字编程	256
20.8	使用套接字传输复杂对象	261
20.9	使用套接字传输流	264
20.10	URL 类	265

20.11 小结	266
20.12 复习题	267
20.13 练习题	267

第五部分 Web 技术

第 21 章 Servlet	271
21.1 servlet 定义	272
21.2 为什么需要 servlet	272
21.3 servlet 的工作原理	273
21.4 建立 Web 环境	274
21.5 安装 Jakarta Tomcat	274
21.5.1 启动 Tomcat	275
21.5.2 测试安装	276
21.6 servlet 编程	277
21.6.1 编写通用 servlet	277
21.6.2 部署 servlet	278
21.6.3 理解 HTTP	281
21.6.4 GET 命令	281
21.6.5 POST 命令	282
21.6.6 其他命令	282
21.6.7 HTTP 1.1	282
21.6.8 HTTP 编程	283
21.8 HTML 表单和 servlet	287
21.9 调用 servlet 的其他类	289
21.10 使用 cookie 维护状态	291
21.11 使用会话对象维护状态	293
21.12 小结	296
21.13 复习题	296
21.14 练习题	296
第 22 章 JavaServer 页面 (JSP)	297
22.1 JSP 是什么, 不是什么	298
22.2 JSP 的优点	298
22.3 JSP 如何工作	299
22.4 JSP 编程	299
22.4.1 表达式	299
22.4.2 向 JSP 页面添加 scriptlet	299
22.4.3 声明	300
22.4.4 添加注释	300
22.4.5 指令	300
22.4.6 隐式对象	300
22.5 部署和运行 JSP	301
22.6 在 JSP 中定义方法	304
22.6.1 条件输出	305

22.6.2 从 HTML 表调用 JSP	306
22.6.3 错误处理	307
22.7 小结	307
22.8 复习题	308
22.9 练习题	308
第 23 章 基于组件的 JavaServer 页面	309
23.1 理解 JSP 操作	311
23.2 如何在 JSP 中包含其他 JSP	312
23.3 include 操作	312
23.4 从 JSP 访问 JavaBean	315
23.5 使用 JavaBean 访问数据库	318
23.5.1 更新数据库	320
23.5.2 从数据库中检索数据	321
23.6 小结	323
23.7 复习题	323
23.8 练习题	323
第 24 章 自定义标记库	324
24.1 什么是自定义标记，什么不是	325
24.2 自定义标记的工作方式	325
24.3 使用自定义标记体	329
24.4 使用自定义标记传递参数	330
24.5 小结	333
24.6 复习题	333
24.7 练习题	334
第 25 章 XML	335
25.1 XML 的重要性	336
25.2 XML 组件	337
25.2.1 XML 文档	337
25.2.2 文档类型定义 (DTD)	338
25.2.3 文档验证	338
25.2.4 分析技术	339
25.3 使用 SAX 分析程序读取 XML 文档	340
25.4 使用文档对象模型 (DOM) 操作 XML 文档	343
25.4.1 在内存中生成 XML 树	343
25.4.2 读取 XML 树	344
25.4.3 输出 XML 树	346
25.4.4 操作 XML 树	346
25.5 XSLT	348
25.5.1 XSL 样式表	349
25.5.2 XSLT 使用示例	350
25.6 使用 JDOM 操作 XML 文档	351
25.6.1 读取 XML 文档	351

25.6.2 JDOM 示例.....	354
25.7 小结.....	355
25.8 复习题.....	355
25.9 练习题.....	356
第 26 章 Web 体系结构.....	357
26.1 组织 Web 站点.....	358
26.1.1 Web 站点的 TCP/IP 层次.....	358
26.1.2 超文本传输协议 (HTTP).....	359
26.1.3 浏览器.....	359
26.1.4 Web 服务器.....	360
26.1.5 应用层.....	360
26.2 Web 设计中的缺陷.....	360
26.3 模型—视图—控制器 (MVC) 架构.....	361
26.4 servlet.....	362
26.5 JavaServer 页面 (JSP).....	363
26.6 JavaScript.....	365
26.7 Enterprise JavaBean.....	365
26.8 Web Services.....	366
26.9 小结.....	368
26.10 复习题.....	368

附 录

附录 A 使用 Borland JBuilder.....	371
A.1 下载 JBuilder.....	372
A.2 安装 JBuilder.....	372
A.3 运行 JBuilder.....	372
A.4 使用 IDE 开发 Java 应用程序.....	374
A.4.1 在项目中创建类.....	374
A.4.2 调试 Java 应用程序.....	376
A.5 使用 IDE 开发 JSP 和 Java Servlet.....	378
A.6 小结.....	381
附录 B Sun One Studio 4 (Forte) 团队版.....	382
B.1 下载 Sun One Studio 4.....	383
B.2 安装 Sun One Studio 4.....	383
B.3 运行 Sun One Studio 4.....	383
B.4 使用 IDE 开发 Java 应用程序.....	385
B.5 创建项目.....	385
B.5.1 在项目中创建类.....	385
B.5.2 调试 Java 应用程序.....	387
B.6 使用 IDE 开发 JSP 和 Java Servlet.....	388
B.6.1 添加 JSP 页面.....	389
B.6.2 添加 Servlet.....	391

B.7 小结.....	392
附录 C 复习题解答.....	393

第一部分 Java 基础

第 1 章 Java 简介

第 2 章 关键字、数据类型和变量

第 3 章 运算符

第 4 章 流控制

第 5 章 方法

第 1 章 Java 简介

在本章中我们将学习如下内容：

- Java 是什么；
- 建立 Java 编程环境；
- 编写第一个 Java 程序。

欢迎使用 Java!

本章将对 Java 语言进行简要的介绍,阐明该语言的所有优点并回答这样一个问题:为什么要使用 Java?本章还将讲述一点 Java 的历史,然后指导您完成第一个 Java 应用程序。

1.1 Java 是什么

Java 是一种具有众多功能的高级编程语言。它可用于开发 Web 页面上的小程序 (applet)、桌面上运行的应用程序、用于客户端和服务器资源通信的服务器端中间件。Java 还可以用于 Web 服务器、嵌入式系统和智能设备中,而这些仅仅是 Java 众多功能的一小部分。在 Web 浏览器端的小程序和 (Web 和企业的) 服务器端应用程序中,都可以找到 Java 的实际应用。

1.1.1 历史

下面是 Java 编程语言的一段简史以及 Java 出现的原因。1991 年, Sun Microsystems 公司 Green 项目的一个研究小组在开发一种控制消费类电子设备的软件,其目标是开发一种编程语言,利用这种语言可对电视机、烘炉,甚至建筑物这样的“智能”设备进行控制并联网,这些设备将能共存并互相通信。

Sun 发布的第一个原型是一种叫作 Star7 的设备,此设备与远程控制类似,可与其他 Star7 设备通信。最初的意图是使用 C++ 来控制 Star7,但却失败了,结果 Green 项目组的成员 James Gosling 开发了一种被称为 Oak 的新语言来控制 Star7。Oak 这个名字来自于 Gosling 在开发这种语言时从办公室窗外看到的一棵树。由于 Oak 这个名字已被使用, Sun 后来就将 Oak 改成了 Java。

由于 Oak 或 Java 是针对小型设备而设计的,因此该语言具有以下与生俱来的优点:

- Java 很小,因此可在内存有限的小型设备上运行。
- Java 高效而且可靠,因此可适用于小型应用——我们都习惯于计算机的系统崩溃,但微波炉的崩溃却令人无法接受。
- Java 可移植,因此可在不同的硬件上运行。

在 1994 年, Green 项目组的成员完全用 Java 开发了一种可以运行 Java 小程序的万维网 (WWW) 浏览器,展示了这种语言的强大功能。该浏览器最初被命名为 WebRunner,而现在则以 HotJava 著称于世。然而,Java 真正焕发出勃勃生机还是在 1995 年,那年, Netscape 公司批准了 Java 编程语言,并在其 Navigator 产品中加入了 Java 的支持。

1997 年, Sun 发布了 Servlet API,这给服务器端的 Web 开发带来了一场革命,使 Web 开发从单一的 CGI 脚本或专用方案中走出。1999 年, Sun 发布了第一版 Java 2 Enterprise Edition (J2EE) 规范,该规范将 JavaServer Pages (JSP) 和 Enterprise JavaBean (EJB) 都纳入一个高度分布式的企业中间件中。从此以后, J2EE 规范与 JSP、Servlet 和 EJB 规范一起发展成一套功能非常强大的程序设计范例。

1.1.2 为什么使用 Java

您或许曾听到办公室中有人谈论 Java,或许您正在学校中学习 Java 课程,或许您已经知道了 Java 是什么,但是不太清楚为什么要使用 Java。无论是哪种情况,以下都是您学习 Java 的理由:

- Java 是一种真正面向对象的编程语言,很适合作为对现实世界进行建模以及解决现实世界中的问题。Java 能够适应最先进的设计方法,并具有可扩展性,因此易于管理大型的项目。
- Java 独立于平台,因此相同的半编译字节代码可以在安装了 Java 虚拟机的 Microsoft Windows、Unix、Apple Macintosh、Linux 以及未来任何杰出的操作系统上运行。
- Java 可以增强 Web 页面的交互性。Java 小程序可提供许多耳熟能详的功能,例如:菜单、数据输入表单以及其他用户界面组件,还可以提供大多数 Web 页面上所出现的一些精美图片和动画。
- 随着 CPU 速度的提高,使用 Java 来构建有效的应用程序变得更为可行。
- Java 中全新的 SWING 库可与 Microsoft Windows 操作系统中的对应控件 (如树控件、选项卡控件、

列表控件、表等)相媲美。

- 在接下来的 10 年中, Java 的前途将不可限量。关于“智能设备”宿主可运行 Java 程序的 Java 虚拟机的讨论越来越多。这些设备, 例如智能烘炉、智能电视、以及智能住宅等都将连成网络, 这样您的电视就可以让 VCR (录像机) 通知烘炉开始烘烤! Java 是这个令人兴奋且不断发展的领域中的前沿技术。

在过去的几年中, Java 已经度过了自己的幼年时代, 实现了概念上的飞跃, 并取得了不可思议的成就。随着它的发展和成熟, Java 将变成未来主流开发语言的这一趋势越来越明显。学习 Java 不仅可以帮助您立刻开发出惊人的 Java 小程序和应用程序, 还可以为将来进行计算机和设备的编程做好准备。

1.1.3 Java 的表征性术语

以下是与 Java 有关的表征性术语。本节将给出这些术语的定义并阐述它们如何适用于 Java。

- 简单;
- 面向对象;
- 分布式;
- 健壮;
- 安全;
- 体系结构无关性;
- 可移植;
- 解释性;
- 多线程。

1. 简单

Java 容易学习也易于实现。其语法类似 C++ 和 C#, 它将所有的标准编程范例以一种易于使用的形式组合起来。实际上, Java 的目标之一就是让开发者能够快速开发和运行程序。此外, Java 具有一致性, 在搞懂了一个问题之后, 您不必担心在遇到类似的问题时会需要不同的解决方案。

2. 面向对象

Java 是一门支持对象、继承和多态的真正面向对象的编程语言, 这使得它非常适合为学习计算机编程奠定良好的基础。我们将从第 6 章“类”开始讨论对象和面向对象程序设计。面向对象的主要好处是可以根据现实生活中的事物(例如苹果、汽车或人物)来构建 Java 程序模型。这种编程范例有助于跨越机器和人类思维方式之间的鸿沟。

3. 分布式

Java 语言自其创建时起, 组件之间的可连接性自创建时起就是 Java 语言的基本原则。在技术发展后, Java 的开发者仍然恪守这一原则。因此, 要开发在多台机器上并发运行和机器间通信的应用程序, 使用 Java 远比使用其他语言简单高效。

4. 健壮

Java 创建之初只是一种为 Web 页面增添交互性并开发简单通信方案的简单机制, 但这已经成为一种遥远的回忆。现在 Java 拥有许多引以为豪的强大功能和特性: 可与最优秀的操作系统相媲美的图形用户界面(GUI); 优于任何其他供应商的 Web 开发方案; 领导市场的分布式企业平台; 一套极为丰富的、可实现数据输入和输出、多线程、高级图像处理、XML 文档操作等功能的库集合。几乎所有能在一门编程语言中找到的功能, 在 Java 中都能找到。

5. 安全

记住, Java 最初就是围绕智能设备之间的通信而设计的。这条原则对 Java 的可用功能和禁用特性作了非常严格的规定。Java 编程语言的各种组件运行在不同的安全环境下。例如, 在 Web 浏览器中运行的小程序不能访问 Web 客户端的底层文件系统, 也不能访问除源站点以外的其他 Web 站点; 分布式组件必须经过授权才能与其他主机进行通讯; 所有的 Java 程序都运行在虚拟机上, 这样能够防止底层操作系统受到侵害。

6. 体系结构无关性

由于 Java 在虚拟机内部（一种采用特定于操作系统的代码编写的程序，对所有的操作系统提供通用的前端）运行，因此完全独立于底层操作系统和硬件。这也是 Java 得以生存的另一项基本原则：能够在任何设备上运行。

7. 可移植

用 Java 开发的程序被编译成由虚拟机指令组成的字节代码，虚拟机会将这些指令翻译成操作系统特定的指令。正因如此，您所编写并编译成字节代码的任何 Java 代码都可以在任何安装有兼容虚拟机的操作系统上运行。因此，Java 的口号是：一次编写，到处运行。业内的玩笑话曾是：一次编写，到处调试。不过现在这种现象已不严重了，特别是在服务器端的领域中。

8. 解释性

Java 是一种解释语言，这意味着 Java 代码只有在执行时才会被编译成机器代码。解释性代码的运行速度往往比完全编译的代码要慢很多，但是 Java 的性能有所改进，因为 Java 代码会被部分编译成字节代码。部分编译功能是 Java 的另一优势，这样，提供给客户的 Java 应用程序就不必是纯粹的、易于阅读的源代码，而可以是二进制代码的形式。

9. 多线程

多线程应用程序能够同时运行多项任务，例如一个多线程应用程序可以在为用户显示图形用户界面的同时从 Internet 上下载股票报价并计算历史平均价格。Java 语言在创建之初便开始设计并完美地实现了线程模型，在本章后面您将会逐渐体会到这一点。

1.2 建立 Java 编程环境

在编译和执行 Java 程序之前，必须安装 Java 编程环境。有各种各样的 Java 编译器，但标准的编译器是 Sun 公司网站上发布的版本。可以从 <http://java.sun.com/j2ee/> 上下载 Java 2 SDK Standard Edition。

在该页上您将看到 Java 最新发布版本、预发布/发布候选版本（如果可用的话）的三种下载类型。这三种下载类型分别是：

- 软件开发工具包（SDK）；
- Java 运行时引擎（JRE）；
- 产品文档。

SDK 用于编译并执行 Java 程序；因此需要下载该工具包。

JRE 用于执行 Java 程序，但不包含任何用于编译的组件。Java 应用程序的使用者必须下载并安装 JRE；但由于 SDK 中包含 JRE，因此不必下载该引擎。

产品文档是专为 SDK 用户准备的。它列出并解释了 Java 中所有的公共类。笔者强烈建议下载并安装产品文档（也可直接在 Internet 上查看）。如果磁盘空间有限，也可在需要时浏览该产品文档。

到编写本书时止，Java 2 标准版的最新版本是 1.4 版。

在下载 SDK 时，会显示指向安装指导的链接，以下章节将概述安装步骤。

1.2.1 Microsoft Windows

Java SDK for Microsoft Windows 是一个 InstallShield® 文件，该文件通过类似向导的屏幕指导您的安装。

- (1) 下载 SDK。
- (2) 运行下载的文件。
- (3) 按提示信息操作。

1.2.2 Linux

针对 Linux 的安装程序有两个版本：自解压的二进制文件，以及包含 RPM 包的二进制文件。

docsriver 文川网
入驻商家 古籍书城

在文川网搜索古籍书城 获取更多电子书

1. 自解压的二进制文件

- (1) 下载 SDK。
- (2) 将该文件复制到 Java 的安装目录。
- (3) 授予文件执行权限：

```
chmod a+x filename.bin
```

- (4) 执行该文件：

```
/filename.bin
```

- (5) 根据屏幕上的指示执行操作。

2. RPM

RPM 文件主要用于 Red Hat Linux 系统中，其中包括了详细的安装说明。RPM 文件非常易于安装，并且几乎不需要用户进行干预。

- (1) 下载 SDK。
- (2) 为文件赋予执行权限：

```
chmod a+x filename.rpm.bin
```

- (3) 执行文件：

```
./filename.rpm.bin
```

- (4) 执行的结果是创建了 a.rpm 文件。如下执行.rpm 文件：

```
rpm -iv filename.rpm
```

1.3 编写第一个 Java 程序

几乎每本书的第一章都会提供经典的“Hello, World”应用程序作为示例。回溯到 20 世纪 80 年代，Brian Kernighan 和 Dennis Ritchie 编写了一本名为 *The C Programming Language* 的书，这本书实际上成了 C 语言编程的典范。在这本书中，他们给出了所有计算机编程书籍在其开篇都应提供的示例：一个在屏幕上显示“Hello, World”的程序。所以，几乎在每本编程书中都可以找到一个标准的“Hello, World”例子。因此，为了不打破传统，清单 1.1（见随书光盘）中就是一段“Hello, World”的 Java 程序。

清单 1.1 HelloWorld.java

1.3.1 编译第一个 Java 程序

在计算机上选择一个专用文件夹，并在该文件夹中创建一个新文件。例如，可以使用 Windows 记事本在下列文件夹中创建一个文件：

```
C:\java
```

在编辑器中输入清单 1.1 中的全部文本并以如下文件名进行保存（注意大小写很重要：“H”和“W”是大写，而其他字母都是小写）：

```
HelloWorld.java
```

如果使用的是记事本，请确保将文件名用引号括起（“HelloWorld.java”），或者在保存后检查文件名，在某些情况下文件名后会被加上.txt 后缀。如果文件名不正确，请改正过来。

打开命令提示符窗口，并转到刚才保存 HelloWorld.java 文件的文件夹。以 Windows 为例，打开命令提示符窗口，如果使用 Windows 95 则是 MS-DOS 提示符窗口，然后输入如下命令：


```
c:  
cd \java
```

这将保证进入 C:驱动器,并将当前目录更改(cd)为 java 子目录。注意在本例中必须将文件保存在该目录下。现在就可以开始编译了。

Java 程序的编译是在命令提示符下用 Java 2 SDK 工具包中的一个工具来完成的: javac, 读作 Java see, 是 Java Compiler 的缩写。可通过向 javac 传递要编译文件的文件名来编译 Java 程序, 例如:

```
javac HelloWorld.java
```

执行这条命令, 将可能看到如下两种结果: 一个新的命令提示符, 或者一组错误信息。笔者的读者和学生在编译 Java 程序时最常见的错误与 Java 所依赖的名为 CLASSPATH 的环境变量有关。Java CLASSPATH 是一个有趣的概念, 这个概念不仅会常常迷惑 Java 编程新手, 同样会让有经验的编程人员犯迷糊。Java CLASSPATH 列出了包含在程序中想要包含的 Java 类的文件夹, 这些文件夹确定了相关类的位置。如果没有定义 CLASSPATH 环境变量, 则其默认值为当前文件夹。如果定义了 CLASSPATH 环境变量, 且希望从当前文件夹编译文件, 则变量值必须明确包括当前文件夹。要检查计算机中 CLASSPATH 环境变量的设置, 请执行如下操作:

```
Windows:  
echo %CLASSPATH%
```

```
Unix:  
echo $CLASSPATH
```

如果结果是空白, 则所收到的任何错误信息都与 CLASSPATH 无关。如果给 CLASSPATH 环境变量指定一个值, 则必须保证包含当前文件夹。在 Windows 下, CLASSPATH 用分号 (;) 分隔各个文件夹, 在 Unix 下则用冒号 (;) 分隔。当前文件夹用句点 (.) 来表示, CLASSPATH 中必须存在句点。要在 CLASSPATH 中添加当前文件夹, 可键入如下内容:

```
Windows:  
Add the current folder to the CLASSPATH  
set CLASSPATH=.;%CLASSPATH%
```

```
Or to erase the CLASSPATH's value  
set CLASSPATH=
```

```
Unix:  
CLASSPATH=.:$CLASSPATH; export CLASSPATH
```

从此以后, 将不会出现与 CLASSPATH 有关的错误。

如果还有其他错误, 请检查以下各项:

- 确保您的 Java 代码与清单 1.1 相符: 所有单词的大小写都必须与清单保持一致, 例如, class 必须小写, “HelloWorld” 的 “H” 和 “W” 必须大写, 而其他字符必须小写。注意, 文件的格式不会影响编译, 例如, 如果将整个程序写在同一行中, 或者在每行之间加入 20 个空行, 都不会对编译造成影响。
- 确保文件名是 HelloWorld.java, 再检查每个字母的大小写, 如果是在 Windows 系统下, 请确保记事本没有在文件名后面加上.txt。可通过使用 dir 命令列出文件所在文件夹的目录来检查文件名(在 Unix 下是 ls 命令)。

```
dir
```

- 确保已经安装了 Java 2 SDK, 并且所下载的是当前版本。要检查这一点, 请在 PATH 下执行以下命令:

```
java -version
```