



Java核心技术

卷 I: 基础知识

(第 11 版 · 英文版)

[美] 凯·S. 霍斯特曼 (Cay S. Horstmann) 著

Core Java

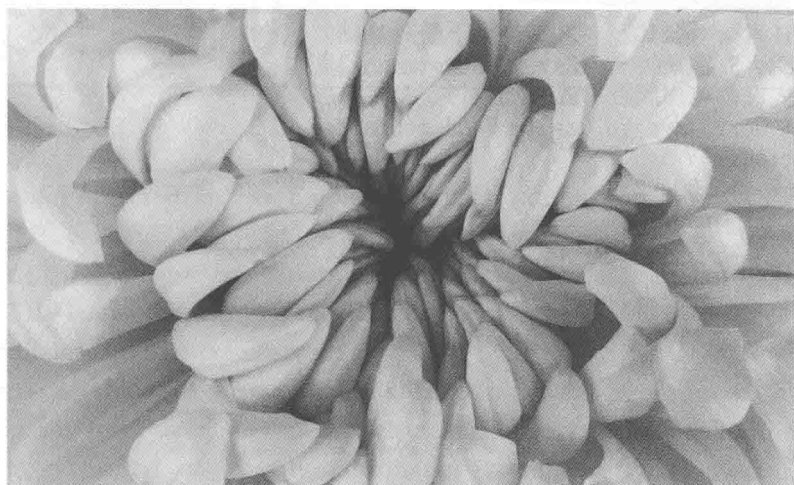
Volume I—Fundamentals, Eleventh Edition



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



Java核心技术

卷I: 基础知识

(第11版·英文版)

[美] 凯·S. 霍斯特曼 (Cay S. Horstmann) 著

Core Java
Volume I—Fundamentals, Eleventh Edition

人民邮电出版社

北京

图书在版编目(CIP)数据

Java核心技术：第11版. 卷I：上下册，基础知识：
英文 / (美) 凯·S. 霍斯特曼 (Cay S. Horstmann) 著
— 北京：人民邮电出版社，2019.5

书名原文：Core Java Volume I—Fundamentals,
Eleventh Edition

ISBN 978-7-115-50492-0

I. ①J… II. ①凯… III. ①JAVA语言—程序设计—
英文 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第285502号

内 容 提 要

本书是经典的《Java 核心技术 卷I：基础知识》的新版。这一版对全书进行了全面更新，以反映 Java SE 9、Java SE 10 和 Java SE 11 的特性。

书中囊括了 Java 的全部基础知识，提供了大量完整且具有实际意义的应用示例，详细介绍了 Java 语言基础、面向对象编程、反射与代理、接口与内部类、事件监听器模型、使用 Swing UI 工具进行图形用户界面程序设计、打包应用程序、异常处理、登录与调试、泛型编程、集合框架、多线程和并发等内容。

本书适合想将 Java 应用于实际项目的软件开发人员、高等院校教师和学生参考阅读。

◆ 著 [美] 凯·S. 霍斯特曼 (Cay S. Horstmann)

责任编辑 杨海玲

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

涿州市京南印刷厂印刷

◆ 开本：800×1000 1/16

印张：54.75

字数：1034 千字

2019 年 5 月第 1 版

印数：1—2 500 册

2019 年 5 月河北第 1 次印刷

著作权合同登记号 图字：01-2018-8409 号

定价：149.00 元（上、下册）

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

版权声明

Authorized Reprint from the English language edition, entitled CORE JAVA VOLUME I--FUNDAMENTALS, 11th Edition by HORSTMANN, CAY S., published by Pearson Education, Inc, publishing as Prentice Hall, Copyright ©2019 Pearson Education Inc. Portions copyright ©1996-2013 Oracle and/or its affiliates. All Rights Reserved.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

ENGLISH language edition published by POSTS AND TELECOM PRESS CO., LTD., Copyright ©2019.

本书英文影印版由 Pearson Education Inc 授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制本书内容。

本书封面贴有 Pearson Education（培生教育）出版集团激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

前言^①

致读者

1995年年底，Java 编程语言异军突起，登上互联网舞台，并且一夜成名。Java 技术旨在成为连接用户与信息的“万能胶”，而不论这些信息来自何处——是 Web 服务器、数据库、信息提供商，还是其他任何可以想到的源头。的确，Java 具备实现这一愿景的得天独厚的条件。Java 是已被广泛接受的、极为稳健的工程语言。其固有的安全性不仅令 Java 程序员放心，也令使用 Java 程序的用户放心。Java 为很多高级编程任务提供了内置支持，比如网络编程、数据库连接和并发。

自 1995 年起，Java 开发工具包（Java Development Kit, JDK）已经发布了 11 个主要版本。在过去的 20 年中，应用编程接口（Application Programming Interface, API）从大约 200 个类发展到 4000 多个。API 现在涉及诸如用户界面构建、数据库管理、国际化、安全和 XML 处理等多个不同领域。

本书是《Java 核心技术》一书第 11 版的卷 I。本书的每一版都紧跟一个 JDK 版本，而且我们每一次都会重写，以利用最新的 Java 特性。这一版也做了更新，以反映 Java SE 9、Java SE 10 和 Java SE 11 的特性。

与之前的版本一样，本书的读者定位仍是那些希望将 Java 应用于实际项目的严谨的程序员。本书假设读者是基础扎实的程序员，已掌握 Java 以外的某种编程语言，而且读者不希望书中充斥着玩具式的示例（比如烤面包机、动物园里的动物，甚至“跳动的文字”）。本书不会使用任何那样的示例。我们的目标是让读者充分理解 Java 语言及其库，而不是带给读者一种“已经理解了”的假象。

本书提供了大量的示例代码，用以说明我们所讨论的几乎每一个语言特性和库特性。我们有意让示例代码尽量简单，以便专注于主要问题。不过绝大多数代码既非凭空杜撰，也未偷工减料。对读者编写自己的代码而言，它们是正确的起点。

我们假设读者愿意，甚至是渴望学习 Java 提供的所有高级特性。例如，我们会详细讨论如下内容：

^① 本前言由臧秀涛翻译。

- 面向对象编程;
- 反射与代理;
- 接口与内部类;
- 异常处理;
- 泛型编程;
- 集合类框架;
- 事件监听器模型;
- 图形用户界面设计;
- 并发。

随着 Java 类库的爆炸式增长,一本书已经无法涵盖严谨的程序员需要了解的所有 Java 特性。因此,我们决定将《Java 核心技术》一书分为两卷。现在读者手中的是卷 I,主要介绍 Java 语言的基本概念,以及用户界面编程的基础知识。《Java 核心技术 卷 II:高级特性》会更进一步,介绍企业级特性和高级用户界面编程,将详细讨论如下内容:

- Stream API;
- 文件处理与正则表达式;
- 数据库;
- XML 处理;
- 注解 (Annotation);
- 国际化;
- 网络编程;
- 高级 GUI 组件;
- 高级图形编程;
- 原生方法。

写书过程中难免会有错误和不准确之处。我们非常想知道书中存在的问题。当然,我们也不希望一个问题被提交多次。为此我们在 <http://horstmann.com/corejava> 上创建了一个页面,存放常见问题 (FAQ)、错误更正和解决方案。在错误提交页面上,我们希望读者先阅读已经提交的错误,因此把报告错误和提供改进建议的表格放到了最后。如果我们没能回答每一个疑问,或没能及时回复,也请不要失望。我们肯定会阅读所有来信,并感谢读者的建议,它们会使本书将来的版本更清晰、更有指导价值。

本书概览

第 1 章给出 Java 有别于其他编程语言的功能的概述。这一章将介绍 Java 语言设计者的初衷，以及成效如何，然后简单回顾这门语言诞生和发展的历史。

第 2 章介绍如何下载与安装 JDK 以及本书的示例代码。然后引导读者使用纯 JDK、支持 Java 的文本编辑器和 Java IDE（集成开发环境）来编译并运行 3 个典型的 Java 程序——一个控制台应用、一个图形界面应用和一个 Applet。

第 3 章开始讨论 Java 语言。这一章涵盖变量、循环和简单函数等基础知识。如果读者是 C 或 C++ 程序员，学起来会很容易，因为这些语言特性的语法和 C 语言的基本一样。如果读者没有 C 语言背景，比如只熟悉 Visual Basic 等语言，则应该仔细阅读这一章。

面向对象编程（Object-Oriented Programming, OOP）是当今主流的编程方式，而 Java 正是一种面向对象编程语言。第 4 章介绍封装（面向对象技术的两个基本构建块之一），以及 Java 语言实现封装的机制，也就是类和方法。除了 Java 语言的语法规则之外，我们还会就如何进行合理的 OOP 设计给出一些建议。这一章最后会介绍奇妙的 javadoc 工具，它可以将代码注释格式化为一组通过超链接关联起来的网页。如果读者熟悉 C++，可以快速浏览一下这一章。如果读者没有面向对象编程背景，在进一步学习 Java 之前，应该花点儿时间掌握 OOP 概念。

类和封装只是 OOP 的一部分，第 5 章将介绍另一个重要概念——继承。继承使我们可以利用现有的类，并根据自己的需求加以修改。这是基本的 Java 编程技术。Java 的继承机制与 C++ 非常像。再强调一次，C++ 程序员去关注两种语言的不同之处即可。

第 6 章介绍如何使用 Java 的接口。与第 5 章的简单继承模型相比，接口让我们更进一步。掌握了接口，就可以充分发挥 Java 面向对象编程的威力了。在介绍完接口之后，我们将探讨 Lambda 表达式，它可以简洁地表达可能会在未来的某个时间点执行的代码块。之后我们将介绍 Java 中一个有用的技术特性——内部类。

第 7 章探讨异常处理。良好的程序也会出现问题，而异常处理就是 Java 用来处理这种状况的健壮性机制。异常处理提供了一种将正常处理代码与错误处理分离的有效手段。当然，即使程序处理了所有异常条件，仍然有可能无法按预期方式工作。这一章最后会给出一些有用的调试技巧。

第 8 章概要讲解泛型编程。泛型编程使程序可读性更好，也更安全。这一章介绍如何使用强类型，以及如何去掉难看且不安全的强制类型转换，再就是如何处理为兼容旧的 Java 版本而带来的复杂问题。

第 9 章的主题是 Java 平台的集合类框架。当需要将多个对象收集到一起，并在之后检索时，应该选择最适合问题环境的集合，而不是将所有元素一股脑儿地扔到一个数组中。这一章将阐述如何使用 Java 预先设计好的标准集合类。

第 10 章开始介绍 GUI 编程。这一章将介绍如何创建窗口，如何在窗口中绘图，如何绘制几何图形，如何以多种字体格式化文本，以及如何显示图片。接下来，我们将介绍如何编写代码来响应事件，比如点击鼠标或者按下键盘上的键。

第 11 章将详细探讨 Swing GUI 工具包。利用 Swing 工具包，可以构建跨平台的图形用户界面。这一章将介绍与各种按钮、文本组件、边框、滑块、列表框、菜单和对话框等控件相关的知识。不过有些更高级的组件将在卷 II 中讨论。

第 12 章是最后一章，探讨并发。该特性支持多个程序任务并行完成。在大部分处理器都有多个核心的时代，我们想让这些核心高效运转，而并发就是 Java 技术的一个重要且激动人心的应用。

本书还附赠了一个讲解 JavaFX 的章节，快速介绍了 JavaFX 这一用于桌面应用的现代 GUI 工具包。读者可以从本书配套网站 <http://horstmann.com/corejava> 下载这一章。

附录列出了 Java 语言的保留字。

排版约定

和很多计算机图书一样，我们使用等宽字体表示计算机代码。



NOTE (注解): 注解会使用这样的图标标记。



TIP (提示): 提示会使用这样的图标标记。



CAUTION (警告): 如果使用不当会存在危险, 我们会使用这样的图标警示。



C++ NOTE (C++注解): 书中有很多 C++注解, 用于解释 Java 和 C++ 的差别。如果没有 C++ 背景, 或者虽有相关背景, 但却视之如梦魇、不愿想起, 都可以跳过。

Java 提供了一个规模庞大的编程库, 或者说 API。在第一次使用某个 API 调用时, 我们会在该节末尾添加一个简短的描述。这些描述不那么正式, 不过我们希望它们比官方的在线 API 文档更有指导意义。类名、接口名或方法名后的数字表示该特性是在 JDK 的哪个版本中引入的。如下所示:

Application Programming Interface 9

程序的源代码都在本书的配套网站上, 在书中它们会以代码清单形式列出。例如:

Listing 1.1 InputTest/InputTest.java

示例代码

本书的配套网站 <http://horstmann.com/corejava> 提供了书中所有的示例代码^①。如何安装 JDK, 以及如何使用示例代码, 详见第 2 章。

^① 本书示例代码也可以从异步社区 (<https://www.epubit.com>) 本书页面下载。

致谢

写一本书需要投入大量精力，修订改版看上去好像容易一些，其实并非如此，尤其是 Java 技术还在持续更新。一本书能成功面世，凝聚了很多人的心血。在此衷心感谢整个《Java 核心技术》团队的贡献。

Pearson 出版社的很多人同样居功至伟，却甘居幕后。我希望他们都能知道我是多么感激他们。和往常一样，衷心感谢我的编辑 Greg Doench，他掌控着本书从编写到出版的整个流程，让我觉察不到幕后的那些人。非常感谢 Julie Nahil 在图书制作方面的支持，感谢 Dmitry Kirsanov 和 Alina Kirsanova 完成书稿的编辑和排版。还要感谢本书早期版本的合著者 Gary Cornell，如今他已开始别的事业。

感谢早期版本的许多读者，他们指出了很多令人尴尬的错误，提供了很多深思熟虑的改进建议。特别感谢非常优秀的审稿团队，他们仔细审读了手稿，使我减少了很多令人尴尬的错误。

本书及早期版本的审阅专家有：Chuck Allison（犹他谷州立大学）、Lance Andersen（甲骨文）、Paul Anderson（Anderson Software Group）、Alec Beaton（IBM）、Cliff Berg、Andrew Binstock（甲骨文）、Joshua Bloch、David Brown、Corky Cartwright、Frank Cohen（PushToTest）、Chris Crane（devXsolution）、Dr. Nicholas J. De Lillo（曼哈顿大学）、RakeshDhoopar（甲骨文）、David Geary（Clarity Training）、Jim Gish（甲骨文）、Brian Goetz（甲骨文）、Angela Gordon、Dan Gordon（Electric Cloud）、Rob Gordon、John Gray（哈特福大学）、Cameron Gregory（olabs.com）、Marty Hall（coreservlets.com 公司）、Vincent Hardy（Adobe 公司）、Dan Harkey（圣何塞州立大学）、William Higgins（IBM）、Vladimir Ivanovic（PointBase）、Jerry Jackson（CA 科技）、Tim Kimmet（沃尔玛公司）、Chris Laffra、Charlie Lai（苹果公司）、Angelika Langer、Doug Langston、Hang Lau（麦吉尔大学）、Mark Lawrence、Doug Lea（纽约州立大学奥斯威戈分校）、Gregory Longshore、Bob Lynch（Lynch Associates）、Philip Milne（咨询师）、Mark Morrissey（俄勒冈研究院）、Mahesh Neelakanta（佛罗里达州大西洋大学）、Hao Pham、Paul Philion、Blake Ragsdell、Stuart Reges（亚利桑那大学）、Simon Ritter（Azul Systems）、Rich Rosen（Interactive Data Corporation）、Peter Sanders（法国尼斯的 ESSI 大学）、Dr. Paul Sanghera（圣何塞州立大学和布鲁克斯学院）、Paul Sevinc（Teamup AG）、

Devang Shah (Sun 微系统公司)、Yoshiki Shibata、Bradley A. Smith、Steven Stelting (甲骨文)、Christopher Taylor、Luke Taylor (Valtech)、George Thiruvathukal、Kim Topley (StreamingEdge)、Janet Traub、Paul Tyma (咨询师)、Peter van der Linden、Christian Ullenboom、Burt Walsh、Dan Xu (甲骨文) 和 John Zavgren (甲骨文)。

Cay Horstmann

2018 年 6 月于加州旧金山

目录

Chapter 1: An Introduction to Java / Java 概述	1
1.1 Java as a Programming Platform / Java 程序设计平台	1
1.2 The Java “White Paper” Buzzwords / Java “白皮书” 中的口号	2
1.2.1 Simple / 简单	3
1.2.2 Object-Oriented / 面向对象	4
1.2.3 Distributed / 分布式	4
1.2.4 Robust / 健壮	4
1.2.5 Secure / 安全	5
1.2.6 Architecture-Neutral / 体系结构中立	6
1.2.7 Portable / 可移植	6
1.2.8 Interpreted / 解释型	7
1.2.9 High-Performance / 高性能	7
1.2.10 Multithreaded / 多线程	8
1.2.11 Dynamic / 动态	8
1.3 Java Applets and the Internet / Java Applet 与 Internet	9
1.4 A Short History of Java / Java 简史	10
1.5 Common Misconceptions about Java / 对 Java 的常见误解	13
Chapter 2: The Java Programming Environment / Java 编程环境	17
2.1 Installing the Java Development Kit / 安装 Java 开发包 (JDK)	18
2.1.1 Downloading the JDK / 下载 JDK	18
2.1.2 Setting up the JDK / 设置 JDK	20
2.1.3 Installing Source Files and Documentation / 安装源文件和文档	22
2.2 Using the Command-Line Tools / 使用命令行工具	23
2.3 Using an Integrated Development Environment / 使用集成开发环境	29
2.4 JShell	32

Chapter 3: Fundamental Programming Structures in Java /

Java 的基本编程结构	37
3.1 A Simple Java Program / 一个简单的 Java 程序	38
3.2 Comments / 注释	41
3.3 Data Types / 数据类型	42
3.3.1 Integer Types / 整型	43
3.3.2 Floating-Point Types / 浮点型	44
3.3.3 The char Type / char 类型	46
3.3.4 Unicode and the char Type / Unicode 与 char 类型	47
3.3.5 The boolean Type / boolean 类型	48
3.4 Variables and Constants / 变量和常量	48
3.4.1 Declaring Variables / 声明变量	48
3.4.2 Initializing Variables / 初始化变量	50
3.4.3 Constants / 常量	51
3.4.4 Enumerated Types / 枚举类型	52
3.5 Operators / 运算符	52
3.5.1 Arithmetic Operators / 算术运算符	52
3.5.2 Mathematical Functions and Constants / 数学函数和常量	54
3.5.3 Conversions between Numeric Types / 数值类型之间的转换	56
3.5.4 Casts / 强制类型转换	57
3.5.5 Combining Assignment with Operators / 组合赋值运算符	58
3.5.6 Increment and Decrement Operators / 自增与自减运算符	58
3.5.7 Relational and boolean Operators / 关系与 boolean 运算符	59
3.5.8 Bitwise Operators / 位运算符	60
3.5.9 Parentheses and Operator Hierarchy / 括号与运算符优先级	61
3.6 Strings / 字符串	62
3.6.1 Substrings / 子串	62
3.6.2 Concatenation / 拼接	63

3.6.3	Strings Are Immutable / String是不可变的	63
3.6.4	Testing Strings for Equality / 测试字符串是否相等	65
3.6.5	Empty and Null Strings / 空串与 null 串	66
3.6.6	Code Points and Code Units / 码位与编码单元	66
3.6.7	The String API / String API	68
3.6.8	Reading the Online API Documentation / 阅读在线 API 文档	71
3.6.9	Building Strings / 构建字符串	74
3.7	Input and Output / 输入和输出	75
3.7.1	Reading Input / 读取输入	75
3.7.2	Formatting Output / 格式化输出	78
3.7.3	File Input and Output / 文件输入和输出	83
3.8	Control Flow / 控制流	86
3.8.1	Block Scope / 块作用域	86
3.8.2	Conditional Statements / 条件语句	87
3.8.3	Loops / 循环	91
3.8.4	Determinate Loops / 确定性循环	95
3.8.5	Multiple Selections—The switch Statement / 多重选择: switch 语句	99
3.8.6	Statements That Break Control Flow / 用于跳出控制流的语句	102
3.9	Big Numbers / 大数	105
3.10	Arrays / 数组	108
3.10.1	Declaring Arrays / 声明数组	108
3.10.2	Accessing Array Elements / 访问数组元素	109
3.10.3	The “for each” Loop / “for each” 循环	110
3.10.4	Array Copying / 数组复制	111
3.10.5	Command-Line Parameters / 命令行参数	112
3.10.6	Array Sorting / 数组排序	113
3.10.7	Multidimensional Arrays / 多维数组	116
3.10.8	Ragged Arrays / 不规则数组	120

Chapter 4: Objects and Classes / 对象与类	125
4.1 Introduction to Object-Oriented Programming / 面向对象编程简介	126
4.1.1 Classes / 类	127
4.1.2 Objects / 对象	128
4.1.3 Identifying Classes / 识别类	129
4.1.4 Relationships between Classes / 类之间的关系	129
4.2 Using Predefined Classes / 使用预定义类	131
4.2.1 Objects and Object Variables / 对象与对象变量	132
4.2.2 The LocalDate Class of the Java Library / Java 库中的 LocalDate 类	135
4.2.3 Mutator and Accessor Methods / 更改器方法与访问器方法	138
4.3 Defining Your Own Classes / 定义自己的类	141
4.3.1 An Employee Class / Employee 类	142
4.3.2 Use of Multiple Source Files / 使用多个源文件	145
4.3.3 Dissecting the Employee Class / 分析 Employee 类	146
4.3.4 First Steps with Constructors / 从构造器开始	146
4.3.5 Declaring Local Variables with var / 使用 var 声明局部变量	148
4.3.6 Working with null References / 使用 null 引用	148
4.3.7 Implicit and Explicit Parameters / 隐式参数与显式参数	150
4.3.8 Benefits of Encapsulation / 封装的好处	151
4.3.9 Class-Based Access Privileges / 基于类的访问权限	154
4.3.10 Private Methods / 私有方法	155
4.3.11 Final Instance Fields / final 实例字段	155
4.4 Static Fields and Methods / 静态的字段与方法	156
4.4.1 Static Fields / 静态字段	156
4.4.2 Static Constants / 静态常量	157
4.4.3 Static Methods / 静态方法	158
4.4.4 Factory Methods / 工厂方法	159
4.4.5 The main Method / main 方法	160

4.5	Method Parameters / 方法参数	163
4.6	Object Construction / 对象构建	170
4.6.1	Overloading / 重载	170
4.6.2	Default Field Initialization / 默认字段初始化	171
4.6.3	The Constructor with No Arguments / 无参构造器	172
4.6.4	Explicit Field Initialization / 显式字段初始化	173
4.6.5	Parameter Names / 参数名	174
4.6.6	Calling Another Constructor / 调用另一个构造器	175
4.6.7	Initialization Blocks / 初始化块	175
4.6.8	Object Destruction and the finalize Method / 对象析构与 finalize 方法	180
4.7	Packages / 包	180
4.7.1	Package Names / 包名	181
4.7.2	Class Importation / 导入类	181
4.7.3	Static Imports / 静态导入	183
4.7.4	Addition of a Class into a Package / 将类添加到某个包中	184
4.7.5	Package Access / 包访问权限	187
4.7.6	The Class Path / 类路径	189
4.7.7	Setting the Class Path / 设置类路径	191
4.8	JAR Files / JAR 文件	192
4.8.1	Creating JAR files / 创建 JAR 文件	192
4.8.2	The Manifest / 清单文件	193
4.8.3	Executable JAR Files / 可执行的 JAR 文件	194
4.8.4	Multi-Release JAR Files / 支持多个 Java 版本的 JAR 文件	195
4.8.5	A Note about Command-Line Options / 关于命令行选项的说明	197
4.9	Documentation Comments / 文档注释	198
4.9.1	Comment Insertion / 插入注释	199
4.9.2	Class Comments / 类注释	199
4.9.3	Method Comments / 方法注释	200
4.9.4	Field Comments / 字段注释	201
4.9.5	General Comments / 通用注释	201

4.9.6	Package Comments / 包注释	202
4.9.7	Comment Extraction / 提取注释	203
4.10	Class Design Hints / 类设计建议	204
Chapter 5: Inheritance / 继承		207
5.1	Classes, Superclasses, and Subclasses / 类、超类与子类	208
5.1.1	Defining Subclasses / 定义子类	208
5.1.2	Overriding Methods / 覆盖方法	210
5.1.3	Subclass Constructors / 子类构造器	211
5.1.4	Inheritance Hierarchies / 继承层次	216
5.1.5	Polymorphism / 多态	217
5.1.6	Understanding Method Calls / 理解方法调用	218
5.1.7	Preventing Inheritance: Final Classes and Methods / 阻止继承: final 修饰的类和方法	221
5.1.8	Casting / 强制类型转换	223
5.1.9	Abstract Classes / 抽象类	225
5.1.10	Protected Access / 受保护访问	231
5.2	Object: The Cosmic Superclass / Object: 所有类的超类	232
5.2.1	Variables of Type Object / Object 类型的变量	232
5.2.2	The equals Method / equals 方法	233
5.2.3	Equality Testing and Inheritance / 相等测试与继承	234
5.2.4	The hashCode Method / hashCode 方法	238
5.2.5	The toString Method / toString 方法	241
5.3	Generic Array Lists / 泛型数组列表	248
5.3.1	Declaring Array Lists / 声明数组列表	248
5.3.2	Accessing Array List Elements / 访问数组列表的元素	251
5.3.3	Compatibility between Typed and Raw Array Lists / 类型化和原始数组列表的兼容性	255
5.4	Object Wrappers and Autoboxing / 对象包装器与自动装箱	256
5.5	Methods with a Variable Number of Parameters / 参数数量可变的方法	260
5.6	Enumeration Classes / 枚举类	261
5.7	Reflection / 反射	264