

专门针对Java SE 8 (JDK 8)全面更新

Java: A Beginner's Guide, Sixth Edition



Java 8

编程入门官方教程(第6版)

[美] Herbert Schildt 著
王楚燕 鱼 静 译



清华大学出版社

Java 8 编程入门

官方教程

(第6版)

[美] Herbert Schildt 著

王楚燕 鱼 静 译

清华大学出版社

北 京

Herbert Schildt

Java : A Beginner's Guide, Sixth Edition

EISBN: 978-0-07-180925-2

Copyright © 2014 by McGraw-Hill Education.

All Rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including without limitation photocopying, recording taping, or any database, information or retrieval system, without the prior written permission of the publisher.

This authorized Chinese translation edition is jointly published by McGraw-Hill Education and Tsinghua University Press Limited. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong, Macao SAR and Taiwan.

Copyright © 2015 by McGraw-Hill Education and Tsinghua University Press Limited.

版权所有。未经出版人事先书面许可,对本出版物的任何部分不得以任何方式或途径复制或传播,包括但不限于复印、录制、录音,或通过任何数据库、信息或可检索的系统。

本授权中文简体字翻译版由麦格劳-希尔(亚洲)教育出版公司和清华大学出版社有限公司合作出版。此版本经授权仅限在中华人民共和国境内(不包括香港特别行政区、澳门特别行政区和台湾)销售。

版权©2015 由麦格劳-希尔(亚洲)教育出版公司与清华大学出版社有限公司所有。

北京市版权局著作权合同登记号 图字: 01-2014-7593

本书封面贴有 McGraw-Hill Education 公司防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Java 8 编程入门官方教程: 第 6 版/(美) 施密特(Schildt, H.) 著; 王楚燕, 鱼静 译. —北京: 清华大学出版社, 2015
书名原文: Java : A Beginner's Guide, Sixth Edition
ISBN 978-7-302-38738-1

I. ①J… II. ①施… ②王… ③鱼… III. ①JAVA 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 284272 号

责任编辑: 王 军 李维杰

封面设计: 牛艳敏

责任校对: 成凤进

责任印制: 王静怡

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 39 字 数: 1023 千字

版 次: 2015 年 1 月第 1 版 印 次: 2015 年 1 月第 1 次印刷

印 数: 1~3000

定 价: 69.80 元

产品编号: 061113-01



译者序

Java 从最初发布到现在，已经快 20 年了。如今，Java 依然是众多开发人员的首选，这自然与其不断更新、紧跟时代有关。这是一门语言生命力旺盛的体现。

Java 发展到今天，最新的版本就是本书要介绍的 Java SE 8。历代版本中，有的对上一版做了重大更新，有的只是修修补补。Java 8 属于前一类，其最突出的地方就是引入了 lambda 表达式。lambda 表达式产生的影响深远：不但改变了概念化的编程方式，而且改变了 Java 代码的编写方式。可以预见，Java 开发人员思考问题的现有方式和编码方法必然会受到冲击。但冲击的后果，就是当开发人员选择在合适的地方使用 lambda 表达式后，工作效率会有明显提升。其他一些新特性还包括接口方法的默认实现，以及终将取代 Swing 的 JavaFX 框架，这些全都足以使开发人员欢欣鼓舞。

可见，Java 并没有沉溺于过去的荣耀、固步自封，而是竭力为开发人员考虑，通过版本的更新，使开发人员有机会利用最新、最有价值的思想和理念提高开发效率。所以选择学习 Java 无疑是正确的决定。那么，要如何学习 Java 呢？

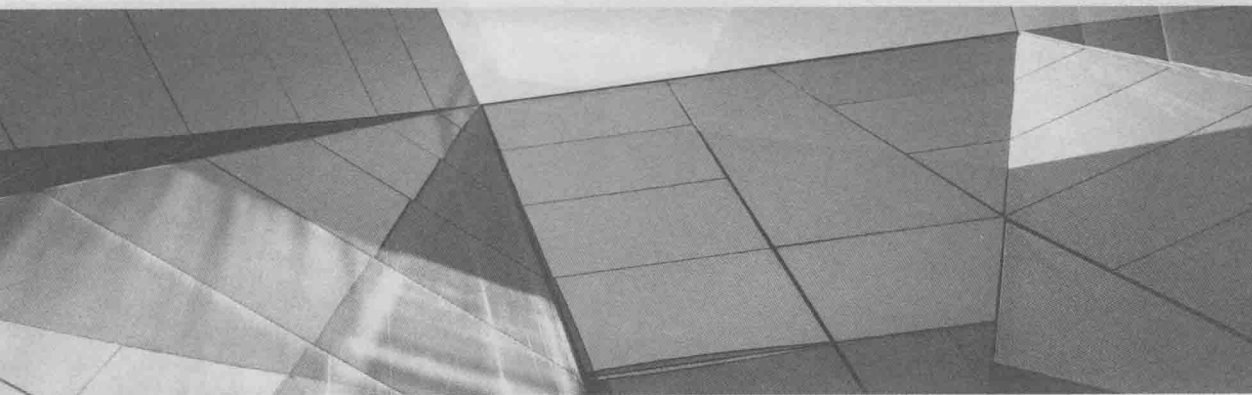
对于入门来说，选择一本教程是最方便、最实际的方法。但是，Java 是一门极受欢迎的语言，市面上介绍 Java 的图书让人眼花缭乱，难道让新手每本都看？事实上，读者既没有必要每本都看，也不会有那样的时间每本都看。那么，选择什么样的图书呢？答案很简单：选择一本优秀的教程。优秀的教程往往由浅入深进行介绍，并且重点突出，内容全面，示例代码丰富且实用性强。作者穿插在特性介绍中的语言，也常常反映出作者丰富的经验和深刻的见解，对于初学者形成正确的认识和良好的编程习惯极有帮助。

本书就是这样一本优秀的教程，用凝练的语言传达最有价值的信息，在 Amazon 网站上

获得了非常好的评价,销量也非常不错。作者 Herbert Schildt 可谓大名鼎鼎,他有几十年的编程经验,也是 Java 语言方面的权威,曾撰写过众多关于 Java 的经典图书。所以本书的质量应该不言而喻。当然,有的作者功力深厚,在写书时不由讲得很深奥,所以作者知名不代表书就一定适合初学者。但是对于本书,作者提到了前提就是读者根本没有编程背景,所以完全从良师的角度,循循善诱,帮助读者从最根本的概念开始,一步一步地学习庞大的 Java 平台。书中除了介绍入门技巧之外,还以“自测题”、“编程练习”和“专家问答”的形式给出了很多实用性的习题和问题,这些都是作者智慧和经验的结晶,仔细研究这些习题和问题会让读者受益匪浅。

这次翻译技术类书籍,最大的体会就是,翻译和自己阅读外文技术书籍差别非常大。虽然阅读外文技术类书籍没有什么障碍,但开始翻译工作之后才真切体会到,想要准确精炼地表达原作者的意思,绝不是自己读懂那么简单。况且本书凝聚了 Java 大师的独到见解,即便专门从事 Java 开发,也没有完全准确地理解并表达所有内容的自信。为了尽可能保证翻译内容的准确完整性,我在翻译过程中,曾多次就不确定的部分征求产品开发团队同事的意见。在此我想对团队的全体同事表达我最真挚的谢意。

在这里要感谢清华大学出版社的编辑,她们为本书的翻译投入了巨大的热情并付出了很多心血。没有你们的帮助和鼓励,本书不可能顺利付梓。本书全部章节由王楚燕、鱼静翻译,参与翻译活动的还有孔祥亮、陈跃华、杜思明、熊晓磊、曹汉鸣、陶晓云、王通、方峻、李小凤、曹晓松、蒋晓冬、邱培强、洪妍、李亮辉、高娟妮、曹小震、陈笑。在此一并感谢!对于这本经典之作,译者本着“诚惶诚恐”的态度,在翻译过程中力求“信、达、雅”,但由于译者水平有限,翻译工作中可能会有不准确的内容,如果读者在阅读过程中所发现的失误和遗漏之处,希望能够多多包涵,并欢迎批评指正。



作者简介

作者简介

畅销书作家 Herbert Schildt 是 Java 语言的权威,他撰写的程序设计图书大约有三十多本,在全世界销售了数百万册,并被翻译成了多种语言。他撰写了众多关于 Java 的图书,包括 *Java: The Complete Reference*、*Herb Schildt's Java Programming Cookbook* 和 *Swing: A Beginner's Guide*,还撰写了一些有关 C、C++ 和 C# 方面的书籍。虽然他对计算机的方方面面都很感兴趣,但主要关注点是计算机语言,包括编译器、解释器和机器人控制语言。他对语言的标准化也有浓厚的兴趣。Schildt 获得了伊利诺伊大学的学士学位。他的咨询中心的电话号码为 (217)586-4683。他的网站为 www.HerbSchildt.com。

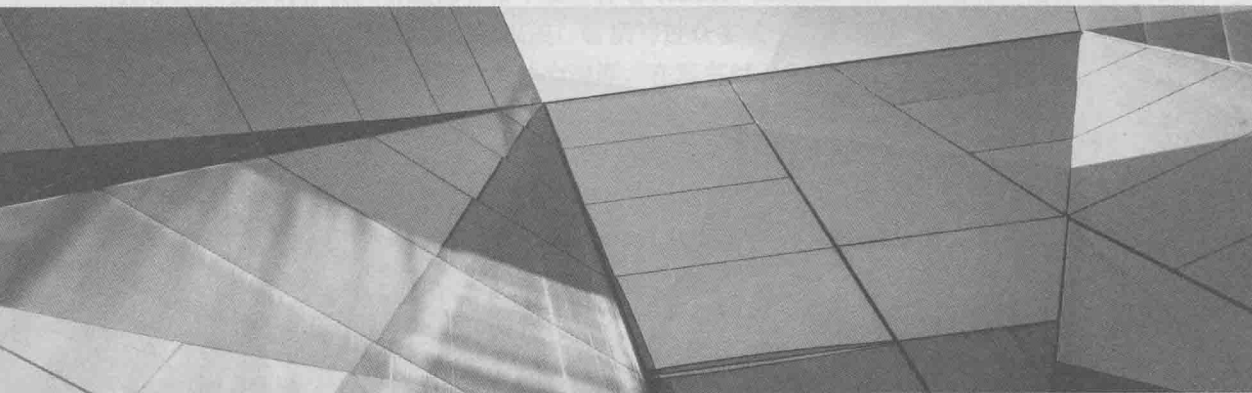
0.1 Java 的发展历史

只有少数几种编程语言对程序设计领域产生了深远的影响,其中,Java 的发展历史尤为引人注目。可以毫不夸张地说,1995 年 Java 公司发布的 Java 1.0 编程语言彻底改变了一场变革,这场变革将网络 Web 转变成了一个高度交互的环境,电脑科学家们开始为网络设计了一个新标准。

多年以来,Java 才刚起步,而 C 和 C++ 和其他语言加入新语言的竞争远胜于 Java。然而,在计算机技术快速发展的今天,随着网络技术的普及,网络环境是它所面临的挑战。Java 已经通过其强大的网络功能,为网络环境提供了一个新的标准。

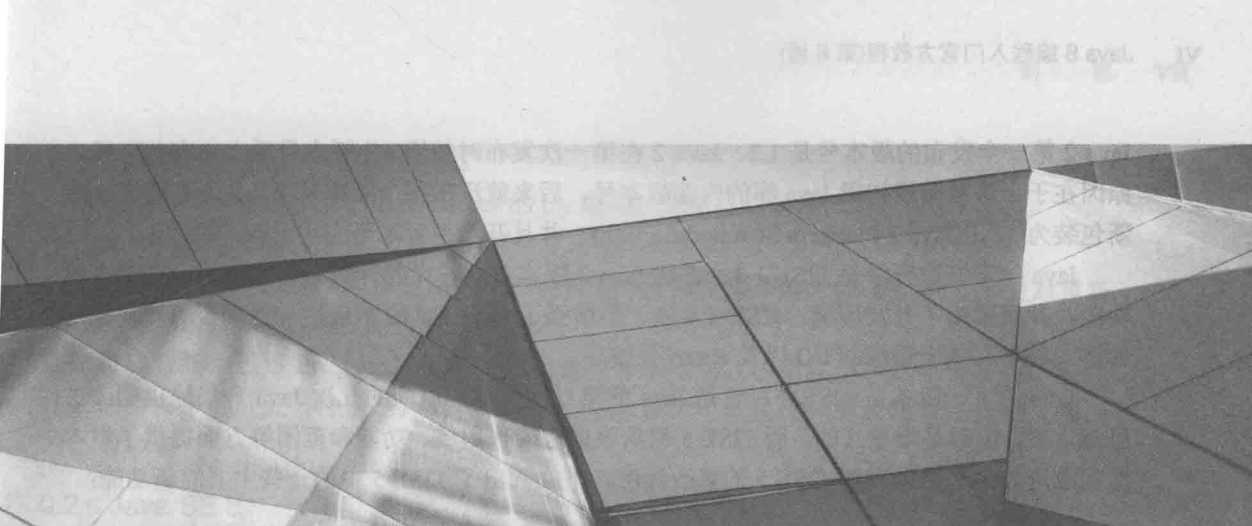
第一个主题的版本是 Java 1.0,它发布的时间比较大,加入了更多的新功能,进行了处理系统的方式,重新配置了 Java 1.0 版本中的许多功能。

第二个主题的版本是 Java 2,它代表了 Java 的第二代。在本书 Java 的“现代化”的到来。



技术编辑简介

Danny Coward 博士在 Java 平台的各个版本上都工作过。他将 Java servlet 的定义引入到 Java EE 平台的第一个版本中，将 Web 服务引入 Java ME 平台，并提出了 Java SE 7 的战略和规划。他开发了 Java FX 技术，最近，他还为 Java EE 7 标准设计了 Java WebSocket API。从用 Java 编写代码到与行业专家设计 API，到他作为 Java Community Process Committee 主管的多年来，他对 Java 技术的多个方面都具有独到的见解。另外，他还是 *Java WebSocket Programming* 一书的作者，并且有一本关于 Java EE 的书也即将出版。Coward 博士拥有牛津大学数学系的学士、硕士和博士学位。



前言

本书旨在帮助你学习 Java 程序设计的基础知识，采用循序渐进的教学方法，安排了许多示例、自测题和编程练习。本书不需要读者具备编程经验，而是从最基础的知识，从如何编译并运行一个 Java 程序开始讲起。然后讨论了构成 Java 语言核心的关键字、功能和结构。还介绍了 Java 的一些最重要的高级功能，如多线程编程和泛型。此外本书还介绍了 Swing 基础和 JavaFX。学完本书后，读者将会牢固地掌握 Java 编程精髓。

值得说明的是，本书只是学习 Java 的起点。Java 不仅仅是一些定义语言的元素，还包括扩展的库和工具来帮助开发程序。要想成为顶尖的 Java 程序员，就必须掌握这些领域。读者在学习完本书之后，就有了足够的知识来继续学习 Java 的其他方面。

0.1 Java 的发展历程

只有少数几种编程语言对程序设计带来过根本性的影响。其中，Java 的影响由于迅速和广泛而格外突出。可以毫不夸张地说，1995 年 Sun 公司发布的 Java 1.0 给计算机程序设计领域带来了一场变革。这场变革迅速地把 Web 转变成了一个高度交互的环境，也给计算机语言的设计设置了一个新标准。

多年以来，Java 不断地发展、演化和修订。和其他语言加入新功能的动作迟缓不同，Java 一直站在计算机程序设计语言的前沿，部分原因是其变革的文化，部分原因是它所面对的变化。Java 已经做过或大或小的多次升级。

第一次主要的升级是 Java 1.1 版，这次升级比较大，加入了很多新的库元素，修订了处理事件的方式，重新配置了 1.0 版本的库中的许多功能。

第二个主要的版本是 Java 2，它代表 Java 的第二代，标志着 Java 的“现代化”的到来。

Java 2 第一个发布的版本号是 1.2。Java 2 在第一次发布时使用 1.2 版本号看上去有些奇怪。原因在于,该号码最初指 Java 库的内部版本号,后来就泛指整个版本号了。Java 2 被 Sun 重新包装为 J2SE(Java 2 Platform Standard Edition),并且开始把版本号应用于该产品。

Java 的下一升级是 J2SE 1.3,它是 Java 2 版本首次较大的升级。它增强了一些已有的功能,并且紧凑了开发环境。J2SE 1.4 进一步增强了 Java。该版本包括一些重要的新功能,如链式异常、基于通道的 I/O 以及 assert 关键字。

Java 的下一版本是 J2SE 5,它是 Java 的第二次变革。以前的几次 Java 升级提供的改进虽然重要,但都是增量式的,而 J2SE 5 却从该语言的作用域、功能和范围等方面提供了根本性的改进。为了帮助理解 J2SE 5 的修改程度,下面列出了 J2SE 5 中的一些主要的新功能:

- 泛型
- 自动装箱/自动拆箱
- 枚举
- 增强型 for-each 形式的 for 循环
- 可变长度实参(varargs)
- 静态导入
- 注解(annotation)

这些条目都是重要的升级,每一个条目都代表了 Java 语言的一处重要改进。其中,泛型、增强型 for 循环和可变长度实参引入了新的语法元素;自动装箱和自动拆箱修改了语法规则;注释增加了一种全新的编程注释方法。

这些新功能的重要性反映在使用的版本号“5”上。从版本号的变化方式看,这一版本的 Java 应该是 1.5。由于新功能和变革如此之多,常规的版本号升级(从 1.4 到 1.5)已无法标识实际的变化,因此 Sun 决定使用版本号 5,以强调发生了重要改进。因此,当前的版本叫做 J2SE 5,开发工具包叫做 JDK 5。但是,为了保持和以前的一致性,Sun 决定使用 1.5 作为内部版本号,也叫做开发版本号。J2SE 5 中的“5”叫做产品版本号。

之后发布的 Java 版本是 Java SE 6, Sun 再次决定修改 Java 平台的名称,把“2”从版本号中删除了。因此,Java 平台现在的名称是 Java SE,官方产品名称是 Java Platform Standard Edition 6,对应的 Java 开发工具包叫做 JDK 6。和 J2SE 5 一样,Java SE 6 中的“6”是指产品的版本号,内部的开发版本号是 1.6。

Java SE 6 建立在 J2SE 5 的基础之上,做了进一步的增强和改进。Java SE 6 并没有对 Java 语言本身添加较大的功能,而是增强了 API 库,添加了多个新包,改进了运行时环境。它在漫长的生命周期(Java 术语)内经历了一些更新,添加了一些升级功能。总之,Java SE 6 进一步巩固了 J2SE 5 建立的领先地位。

Java 的下一版本是 Java SE 7,对应的 Java 开发工具包叫做 JDK 7,内部版本号是 1.7。Java SE 7 是 Oracle 收购 Sun Microsystems 之后发布的第一个主版本。Java SE 7 包含许多新功能,对语言和 API 库做了许多增强。Java SE 7 添加的最重要的功能是在 Project Coin 中开发的那些功能。Project Coin 的目的是确保把对 Java 语言所做的很多小改动包含到 JDK 7 中,其中包括:

- 现在 String 可以控制 switch 语句。
- 二进制整型字面值。

- 在数值面值中使用下划线。
- 新增一种叫做 `try-with-resources` 的 `try` 语句，支持自动资源管理。
- 构造泛型实例时，通过菱形运算符使用类型推断。
- 增强了异常处理，可以使用单个 `catch` 捕获两个或更多个异常(多重捕获)，并且对重新抛出的异常可以进行更好的类型检查。

可以看到，虽然 Project Coin 中的功能被视为小改动，但是“小”这个词实在不能体现它们所带来的好处。特别是，`try-with-resources` 语句对大量代码的编写方式会产生深远的影响。

0.2 Java SE 8

Java 的最新版本是 Java SE 8，对应的开发工具包是 JDK 8，内部的开发版本号是 1.8。JDK 8 表示这是对 Java 语言的一次重大升级，因为本次升级包含了一种意义深远的新语言功能：`lambda` 表达式。`lambda` 表达式的影响深远：不但改变了概念化的编程方式，而且改变了 Java 代码的编写方式。使用 `lambda` 表达式，可以简化并减少创建某个结构所需的源代码量。另外，使用 `lambda` 表达式还可以将新的运算符 `->` 和一种新的语法元素引入到 Java 语言中。`lambda` 表达式有助于确保 Java 成为用户所期望的充满活力且敏捷的语言。

除了 `lambda` 表达式，JDK 8 中还新增了一些其他重要的功能。例如，从 JDK 8 开始，通过接口可以为指定的方法定义默认实现。JDK 8 也捆绑了对 JavaFX、Java 的新 GUI 框架的支持。期待 JavaFX 不久便能够在几乎所有的 Java 应用程序中扮演重要的角色，并且最终代替用于大多数基于 GUI 项目的 Swing。总之，Java SE 8 这一主要版本扩展了 Java 语言的功能，并且改变了 Java 代码的编写方式，带来的影响足够深远，在几年后将体现出来。本书中的内容已更新至 Java SE 8 版本，包括一些新的功能、更新和其他内容。

0.3 本书的组织结构

本书采用教程式的组织结构，每一章都建立在前面的基础之上。本书共分 17 章，每一章讨论一个有关 Java 的主题。本书的特色就在于包含了许多便于读者学习的特色内容。

- 关键技能与概念：每一章都首先介绍一些该章中要介绍的重要技能。
- 自测题：每一章都有自测题，测试读者学习到的知识。答案在附录 A 中提供。
- 专家问答：每一章中都穿插一些“专家问答”，以一问一答的形式介绍补充知识和要点。
- 编程练习：每一章中都包含一或两个编程练习，以帮助读者将学习到的知识应用到实践中去。很多这样的练习都是实际的示例，读者可以将其用作自己的程序的起点。

0.4 本书不需要读者具备编程经验

本书假定读者没有任何编程经验。如果读者没有编程经验，阅读本书是正确的选择。如果读者有过一些编程经验，在阅读本书时可以加快速度。但是要记住，Java 在几个重要的地方与其他一些流行的计算机语言不同，所以不要急于下结论。因此，即使读者是经验丰富的程序员，也仍然建议仔细阅读本书。

0.5 本书需要的软件环境

要想编译和运行本书提供的所有程序，需要获得 Oracle 提供的最新版本的 Java Development Kit (JDK)。在撰写本书时，最新版本为 JDK 8，这是 Java SE 8 使用的 JDK 版本。本书在第 1 章介绍如何获得 Java JDK。

如果读者使用早期版本的 Java，也仍然可以阅读本书，只是无法编译和运行使用了 Java 新功能的程序。

0.6 不要忘记 Web 上的代码

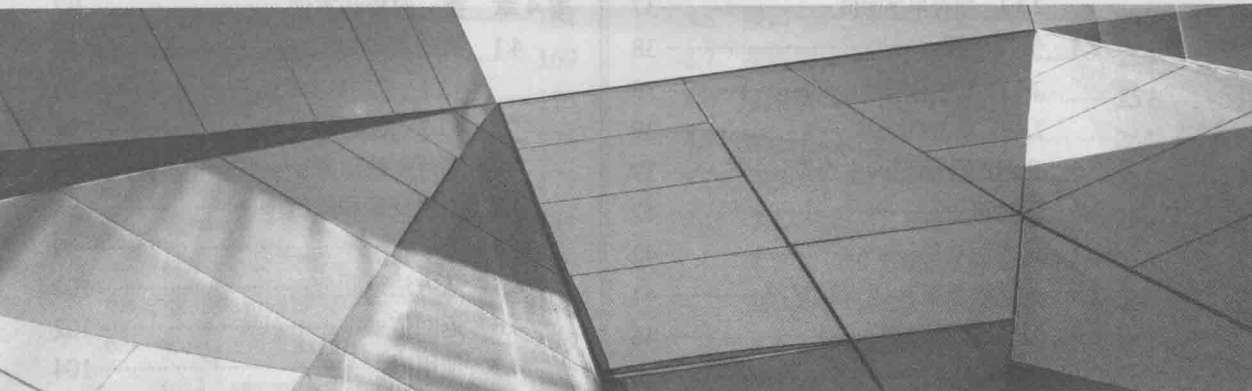
本书所有示例和编程项目的源代码都可以免费从 Web 网址 www.oraclepressbooks.com 和 www.tupwk.com.cn/download 获得。

0.7 特别感谢

特别感谢本书的技术编辑 Danny Coward。他对本书提出了许多宝贵的建议和意见，对此十分感谢。

本书是引导读者进入 Herb Schildt 系列编程图书的大门，对下面的一些书你也会感兴趣：

- Java 8 编程参考官方教程(第 9 版)
- 新手学 JavaFX



目 录

第 1 章 Java 基础	1
1.1 Java 的起源	2
1.1.1 Java 与 C 和 C++ 的关系	3
1.1.2 Java 与 C# 的关系	3
1.2 Java 对 Internet 的贡献	4
1.2.1 Java applet	4
1.2.2 安全性	4
1.2.3 可移植性	5
1.3 Java 的魔法：字节码	5
1.4 Java 的主要术语	6
1.5 面向对象程序设计	7
1.5.1 封装	8
1.5.2 多态性	8
1.5.3 继承	9
1.6 获得 Java 开发工具包	9
1.7 第一个简单的程序	10
1.7.1 输入程序	10
1.7.2 编译程序	11
1.7.3 逐行分析第一个程序	11
1.8 处理语法错误	13
1.9 第二个简单程序	14

1.10 另一种数据类型	16
1.11 两条控制语句	18
1.11.1 if 语句	18
1.11.2 for 循环语句	20
1.12 创建代码块	21
1.13 分号和定位	22
1.14 缩进原则	23
1.15 Java 关键字	25
1.16 Java 标识符	25
1.17 Java 类库	26
第 2 章 数据类型与运算符	29
2.1 数据类型为什么重要	30
2.2 Java 的基本类型	30
2.2.1 整数类型	31
2.2.2 浮点型	32
2.2.3 字符型	33
2.2.4 布尔类型	34
2.3 字面值	36
2.3.1 十六进制、八进制和 二进制字面值	36
2.3.2 字符转义序列	37

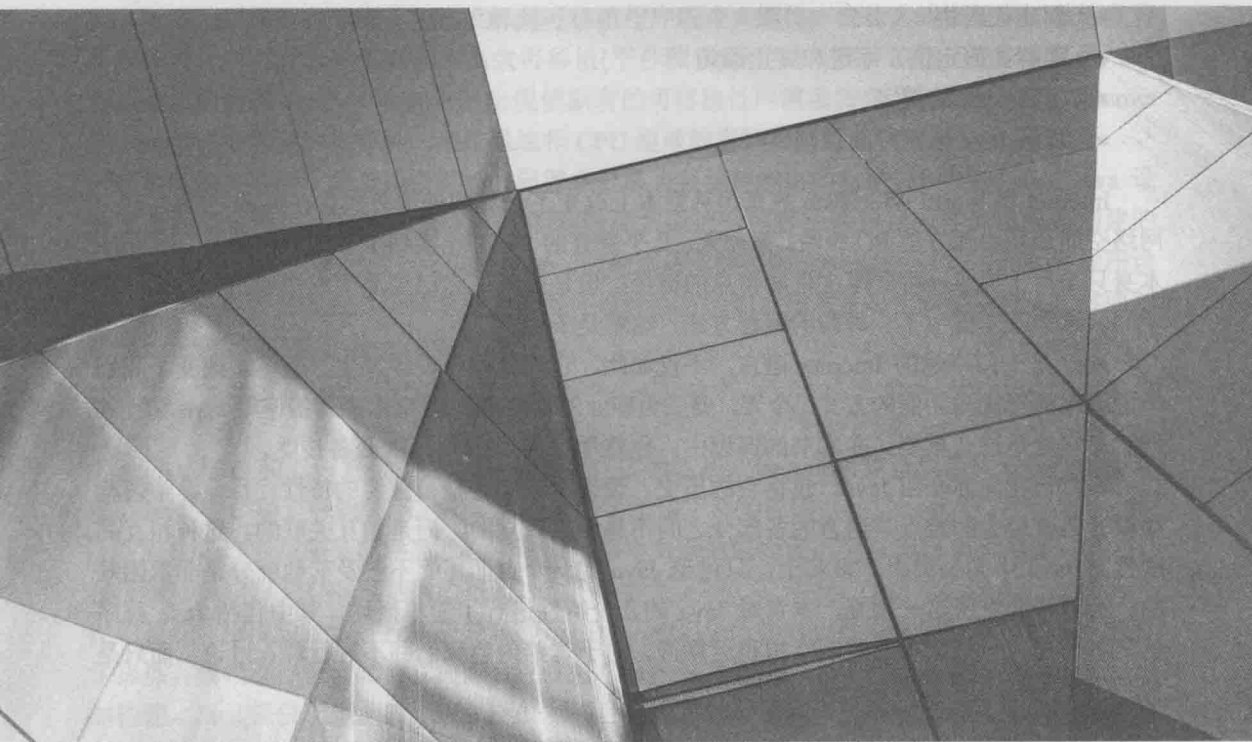
2.3.3 字符串面值	37	第 4 章 类、对象和方法	93
2.4 变量详解	38	4.1 类的基础知识	94
2.4.1 初始化变量	39	4.1.1 类的基本形式	94
2.4.2 动态初始化	39	4.1.2 定义类	95
2.5 变量的作用域和生命期	39	4.2 如何创建对象	98
2.6 运算符	42	4.3 引用变量和赋值	98
2.7 算术运算符	42	4.4 方法	99
2.8 关系运算符和逻辑运算符	44	4.5 从方法返回值	101
2.9 短路逻辑运算符	46	4.6 返回值	102
2.10 赋值运算符	47	4.7 使用形参	104
2.11 速记赋值	47	4.8 构造函数	112
2.12 赋值中的类型转换	48	4.9 带形参的构造函数	113
2.13 不兼容类型的强制转换	50	4.10 深入介绍 new 运算符	115
2.14 运算符优先级	51	4.11 垃圾回收	115
2.15 表达式	53	4.12 this 关键字	119
2.15.1 表达式中的类型转换	53	第 5 章 其他数据类型与运算符	123
2.15.2 间距和圆括号	55	5.1 数组	124
第 3 章 程序控制语句	57	5.2 多维数组	129
3.1 从键盘输入字符	58	5.3 不规则数组	130
3.2 if 语句	59	5.3.1 三维或更多维的数组	131
3.2.1 嵌套 if 语句	60	5.3.2 初始化多维数组	131
3.2.2 if-else-if 阶梯状结构	61	5.4 另一种声明数组的语法	132
3.3 switch 语句	62	5.5 数组引用赋值	133
3.4 for 循环	68	5.6 使用 length 成员	134
3.4.1 for 循环的一些变体	69	5.7 for-each 形式的循环	139
3.4.2 缺失部分要素的 for 循环	70	5.7.1 迭代多维数组	142
3.4.3 无限循环	71	5.7.2 应用增强型 for 循环	143
3.4.4 没有循环体的循环	72	5.8 字符串	144
3.4.5 在 for 循环内部声明循环 控制变量	72	5.8.1 构造字符串	144
3.4.6 增强型 for 循环	73	5.8.2 操作字符串	145
3.5 while 循环	73	5.8.3 字符串数组	147
3.6 do-while 循环	75	5.8.4 字符串是不可变的	148
3.7 使用 break 语句退出循环	79	5.8.5 使用 String 控制 switch 语句	149
3.8 将 break 语句作为 一种 goto 语句使用	81	5.9 使用命令行实参	150
3.9 使用 continue 语句	85	5.10 位运算符	151
3.10 嵌套循环	89	5.10.1 位运算符的与、或、异 或和非	151
		5.10.2 移位运算符	155
		5.10.3 位运算符的赋值速记符	157
		5.11 ?运算符	160

第 6 章 方法和类详解	163
6.1 控制对类成员的访问	164
6.2 向方法传递对象	169
6.3 返回对象	173
6.4 方法重载	174
6.5 重载构造函数	179
6.6 递归	184
6.7 理解 static 关键字	186
6.8 嵌套类和内部类	192
6.9 varargs(可变长度实参)	195
6.9.1 varargs 基础	195
6.9.2 重载 varargs 方法	198
6.9.3 varargs 和歧义	199
第 7 章 继承	203
7.1 继承的基础知识	204
7.2 成员访问与继承	207
7.3 构造函数和继承	209
7.4 使用 super 调用超类构造函数	211
7.5 使用 super 访问超类成员	215
7.6 创建多级层次结构	218
7.7 何时调用构造函数	221
7.8 超类引用和子类对象	222
7.9 方法重写	227
7.10 重写的方法支持多态性	229
7.11 为何使用重写方法	231
7.12 使用抽象类	235
7.13 使用 final	239
7.13.1 使用 final 防止重写	239
7.13.2 使用 final 防止继承	239
7.13.3 对数据成员使用 final	240
7.14 Object 类	241
第 8 章 包和接口	243
8.1 包	244
8.1.1 定义包	244
8.1.2 寻找包和 CLASSPATH	245
8.1.3 一个简短的包示例	245
8.2 包和成员访问	247
8.3 理解被保护的成员	249
8.4 导入包	251

8.5 Java 的类库位于包中	252
8.6 接口	253
8.7 实现接口	254
8.8 使用接口引用	257
8.9 接口中的变量	264
8.10 接口能够被扩展	265
8.11 默认接口方法	266
8.11.1 默认方法的基础知识	266
8.11.2 默认方法的实际应用	268
8.11.3 多继承问题	269
8.12 在接口中使用静态方法	270
8.13 有关包和接口的最后思考	271
第 9 章 异常处理	273
9.1 异常的层次结构	274
9.2 异常处理基础	274
9.2.1 使用关键字 try 和 catch	275
9.2.2 一个简单的异常示例	276
9.3 未捕获异常的结果	277
9.4 使用多个 catch 语句	280
9.5 捕获子类异常	281
9.6 try 代码块可以嵌套	282
9.7 抛出异常	283
9.8 Throwable 详解	285
9.9 使用 finally	286
9.10 使用 throws 语句	288
9.11 新增的 3 种异常功能	289
9.12 Java 的内置异常	291
9.13 创建异常子类	293
第 10 章 使用 I/O	299
10.1 Java 的 I/O 基于流	300
10.2 字节流和字符流	300
10.3 字节流类	301
10.4 字符流类	301
10.5 预定义流	302
10.6 使用字节流	302
10.6.1 读取控制台输入	303
10.6.2 写入控制台输出	304
10.7 使用字节流读写文件	305
10.7.1 从文件输入	305

10.7.2 写入文件.....	309	12.12 注解(元数据).....	393
10.8 自动关闭文件.....	311	第 13 章 泛型.....	397
10.9 读写二进制数据.....	313	13.1 泛型的基础知识.....	398
10.10 随机访问文件.....	317	13.2 一个简单的泛型示例.....	399
10.11 使用 Java 字符流.....	319	13.2.1 泛型只能用于引用类型.....	402
10.11.1 使用字符流的 控制台输入.....	320	13.2.2 泛型类型是否相同基于 其类型实参.....	402
10.11.2 使用字符流的 控制台输出.....	323	13.2.3 带有两个类型形参的 泛型类.....	402
10.12 使用字符流的文件 I/O.....	324	13.2.4 泛型类的一般形式.....	404
10.12.1 使用 FileWriter.....	324	13.3 约束类型.....	404
10.12.2 使用 FileReader.....	325	13.4 使用通配符实参.....	407
10.13 使用 Java 的类型封装器转换 数值字符串.....	326	13.5 约束通配符.....	410
第 11 章 多线程程序设计.....	337	13.6 泛型方法.....	413
11.1 多线程的基础知识.....	338	13.7 泛型构造函数.....	415
11.2 Thread 类和 Runnable 接口.....	339	13.8 泛型接口.....	416
11.3 创建一个线程.....	339	13.9 原类型和遗留代码.....	422
11.4 创建多个线程.....	346	13.10 使用菱形运算符进行 类型推断.....	425
11.5 确定线程何时结束.....	348	13.11 擦除特性.....	426
11.6 线程的优先级.....	351	13.12 歧义错误.....	426
11.7 同步.....	354	13.13 一些泛型限制.....	427
11.8 使用同步方法.....	354	13.13.1 类型形参不能实例化.....	427
11.9 同步语句.....	357	13.13.2 对静态成员的限制.....	428
11.10 使用 notify()、wait()和 notifyAll() 的线程通信.....	360	13.13.3 泛型数组限制.....	428
11.11 线程的挂起、继续执行和停止.....	365	13.13.4 泛型异常限制.....	429
第 12 章 枚举、自动装箱、静态 导入和注释.....	371	13.14 继续学习泛型.....	429
12.1 枚举.....	372	第 14 章 lambda 表达式和方法引用.....	431
12.2 Java 语言中的枚举是类类型.....	374	14.1 lambda 表达式简介.....	432
12.3 values()和 valueOf()方法.....	374	14.1.1 lambda 表达式的基础知识.....	432
12.4 构造函数、方法、实例变量 和枚举.....	376	14.1.2 函数式接口.....	433
12.5 枚举继承 enum.....	378	14.1.3 几个 lambda 表达式示例.....	435
12.6 自动装箱.....	384	14.2 块 lambda 表达式.....	440
12.7 类型封装器.....	385	14.3 泛型函数式接口.....	441
12.8 自动装箱的基础知识.....	386	14.4 lambda 表达式和变量捕获.....	447
12.9 自动装箱和方法.....	387	14.5 从 lambda 表达式中抛出异常.....	448
12.10 发生在表达式中的自动装箱/ 自动拆箱.....	388	14.6 方法引用.....	449
12.11 静态导入.....	390	14.6.1 静态方法的方法引用.....	449
		14.6.2 实例方法的方法引用.....	451
		14.7 构造函数引用.....	455
		14.8 预定义的函数式接口.....	457

第 15 章	applet、事件和其他主题	461	16.4	第一个简单的 Swing 程序	492
15.1	applet 的基础知识	462	16.5	使用 JButton	497
15.2	applet 的组织和基本构件	465	16.6	使用 JTextField	500
15.3	applet 架构	465	16.7	使用 JCheckBox	504
15.4	完整的 applet 框架	465	16.8	使用 JList	507
15.5	applet 的初始化与终止	467	15.9	使用匿名内部类或 lambda 表达式 来处理事件	515
15.6	请求重绘	467	16.10	创建 Swing applet	517
15.7	使用状态窗口	472	第 17 章	JavaFX 简介	521
15.8	向 applet 传递形参	473	17.1	JavaFX 的基本概念	522
15.9	Applet 类	474	17.1.1	JavaFX 包	522
15.10	事件处理	476	17.1.2	Stage 和 Scene 类	523
15.11	委派事件模型	476	17.1.3	节点和场景图	523
15.12	事件	476	17.1.4	布局	523
15.12.1	事件源	476	17.1.5	Application 类和生命 周期方法	523
15.12.2	事件侦听器	477	17.1.6	启动 JavaFX 应用程序	524
15.12.3	事件类	477	17.2	JavaFX 应用程序的骨架	524
15.12.4	事件侦听器接口	478	17.3	编译和运行 JavaFX 程序	527
15.13	使用委派事件模型	479	17.4	应用程序线程	527
15.13.1	处理鼠标事件和鼠标移动 事件	479	17.5	使用简单的 JavaFX 控件 Label	528
15.13.2	一个简单的鼠标事件 applet	480	17.6	使用按钮和事件	530
15.14	其他 Java 关键字	482	17.6.1	事件基础	530
15.14.1	transient 和 volatile 修饰符	483	17.6.2	按钮控件简介	531
15.14.2	instanceof	483	17.6.3	演示事件处理和按钮	531
15.14.3	strictfp	483	17.7	其他 3 个 JavaFX 控件	534
15.14.4	assert	483	17.7.1	CheckBox	534
15.14.5	native 方法	484	17.7.2	ListView	538
第 16 章	Swing 介绍	487	17.7.3	TextField	543
16.1	Swing 的起源和设计原则	488	17.8	效果和变换简介	546
16.2	组件和容器	490	17.8.1	效果	546
16.2.1	组件	490	17.8.2	变换	548
16.2.2	容器	491	17.8.3	演示效果和变换	549
16.2.3	顶级容器窗格	491	17.9	进一步学习	552
16.3	布局管理器	491	附录 A	自测题答案	555
			附录 B	使用 Java 的文档注释	599



第1章

Java 基础

关键技能与概念

- 了解 Java 的历史和基本原理
- 理解 Java 对 Internet 的贡献
- 理解字节码的重要性
- 了解 Java 的术语
- 理解面向对象程序设计的基本原理
- 创建、编译和运行一个简单的 Java 程序
- 使用变量
- 使用 if 和 for 控制语句