

Java语言导学

雷蒙德·盖拉多 (Raymond Gallardo)

斯科特·霍梅尔 (Scott Hommel)

[美]

索亚·坎南 (Sowmya Kannan)

著

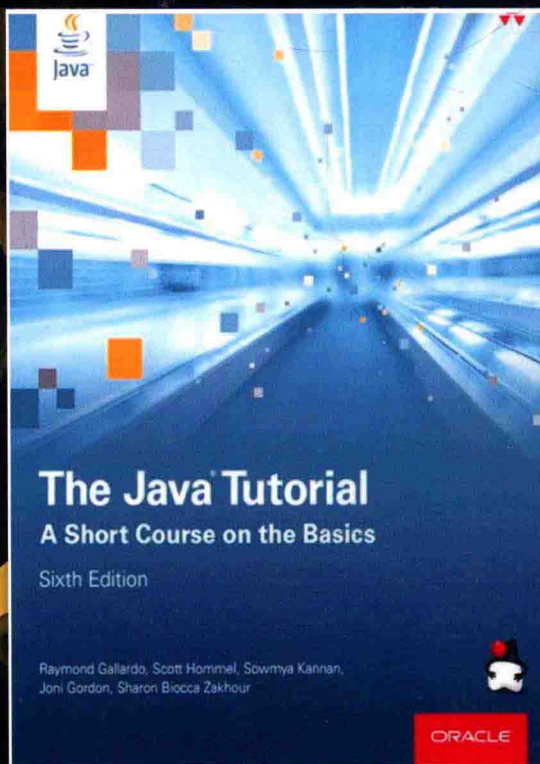
琼尼·戈登 (Joni Gordon)

沙伦·比奥卡·扎卡沃 (Sharon Biocca Zakhour)

董笑菊 薛建新 吴帆 译

The Java Tutorial

A Short Course on the Basics Sixth Edition



计 算 机 科 学 丛 书

原书第6版

Java语言导学

雷蒙德·盖拉多 (Raymond Gallardo)

斯科特·霍梅尔 (Scott Hommel)

[美]

索亚·坎南 (Sowmya Kannan)

著

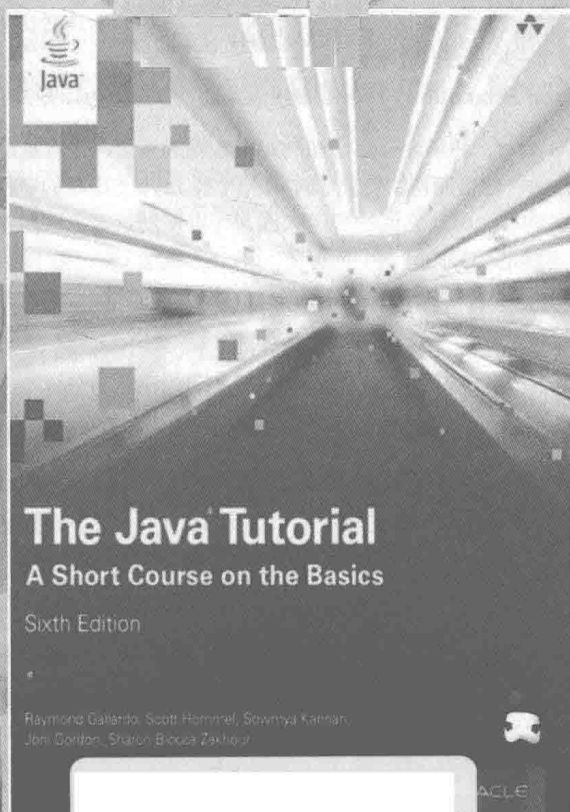
琼尼·戈登 (Joni Gordon)

沙伦·比奥卡·扎卡沃 (Sharon Biocca Zakhour)

董笑菊 薛建新 吴帆 译

The Java Tutorial

A Short Course on the Basics Sixth Edition



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

Java 语言导学 (原书第 6 版)/(美) 雷蒙德·盖拉多 (Raymond Gallardo) 等著; 董笑菊, 薛建新, 吴帆译. —北京: 机械工业出版社, 2017.7
(计算机科学丛书)

书名原文: The Java Tutorial: A Short Course on the Basics, Sixth Edition

ISBN 978-7-111-57330-2

I. J… II. ①雷… ②董… ③薛… ④吴… III. JAVA 语言—程序设计 IV. TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 142775 号

本书版权登记号: 图字: 01-2015-1930

Authorized translation from the English language edition, entitled *The Java Tutorial: A Short Course on the Basics, Sixth Edition*, 978-0-13-403408-9 by Raymond Gallardo, Scott Hommel, Sowmya Kannan, Joni Gordon, Sharon Biocca Zakhour, published by Pearson Education, Inc., Copyright © 2015.

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.

Chinese simplified language edition published by Pearson Education Asia Ltd., and China Machine Press Copyright © 2017.

本书中文简体字版由 Pearson Education (培生教育出版集团) 授权机械工业出版社在中华人民共和国境内 (不包括香港、澳门特别行政区及台湾地区) 独家出版发行。未经出版者书面许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

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

本书基于 Java SE 8 编写, 清晰地介绍了面向对象编程的概念、语言基础等, 涵盖了 Lambda 表达式、类型注解、默认方法、聚合操作、日期/时间 API 等 Java SE 8 的新特性, 更新了限制恶意小程序的攻击和 RIA 等相关内容。每章后面的问题和练习可以帮助读者巩固所学知识, 此外, 附录中还给出 Java 编程语言认证考试的相关内容。

本书可作为高等院校计算机等相关专业的基础教材, 也可作为编程初学者和中级程序员的参考资料。

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 唐晓琳

印 刷: 北京瑞德印刷有限公司

开 本: 185mm×260mm 1/16

书 号: ISBN 978-7-111-57330-2

责任校对: 殷 虹

版 次: 2017 年 7 月第 1 版第 1 次印刷

印 张: 35

定 价: 99.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

购书热线: (010) 68326294 88379649 68995259

投稿热线: (010) 88379604

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

文艺复兴以来，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的优势，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅肇划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自1998年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson, McGraw-Hill, Elsevier, MIT, John Wiley & Sons, Cengage 等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出 Andrew S. Tanenbaum, Bjarne Stroustrup, Brian W. Kernighan, Dennis Ritchie, Jim Gray, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Abraham Silberschatz, William Stallings, Donald E. Knuth, John L. Hennessy, Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力相助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专门为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

华章网站：www.hzbook.com

电子邮件：hzjsj@hzbook.com

联系电话：(010) 88379604

联系地址：北京市西城区百万庄南街1号

邮政编码：100037



华章科技图书出版中心

2015年3月，我们完成了《The Java Tutorial, Fifth Edition》的全部翻译和校正工作。非常幸运，原书第5版译文赶在Java 25周年庆的档期出版。更为欣慰的是，我们的翻译质量得到了市场和出版社的肯定——原书第5版译文出版5个月后，朱劼编辑再次联系我们参与原书第6版的修订和翻译工作。

原书第6版由Oracle公司的资深技术作家根据Java SE 8版本及其API编写而成，是Java SE 8发布后不久就完成的相关教程，也是初级和中级Java程序员学习Java SE 8和进行Java考试认证的权威资料。截至出版时，Java SE 8程序员II级考试OCP Java SE 8程序员认证的考试大纲尚未发布。

相对于原书第5版，第6版不仅在例子上有较大调整，而且新增许多Java SE 8的新特性，如在面向对象程序语言中引入函数式程序语言的组件，给出日期和时间的综合模型等。第6版根据Java SE 8对内容进行大量修订，并增加章节介绍这些新特性。新增的主要内容包括Lambda表达式、类型注解、可插拔类型系统、聚合操作、默认方法、日期和时间API等。另外，章节内容也进行了调整，各章都修订了不少内容，这里不一一列举，详情可查阅目录。

翻译第5版之前，我们曾经调查过对不同译著的评论，总结出译文被诟病的两个方面：多直译，少意译，译文表述不符合中文的表述习惯；专业术语前后不一致。和原书第5版的翻译原则一样，翻译第6版时，我们在尊重技术和原文的基础上，尽可能关注这两个方面。对于发现的原书第5版的错误，也在译稿中做了修正。

本书翻译是在原书第5版译文基础上完成的，参加本书翻译的老师有：上海交通大学计算机科学与工程系董笑菊博士（第8～13章）、上海第二工业大学计算机与信息工程学院薛建新博士（第1～7、21、22章以及附录）、上海交通大学计算机科学与工程系吴帆博士（第14～20章）。虽然我们已经对译稿进行仔细校对和整合，使得译文尽可能符合中文习惯和保持术语的一致性，但鉴于水平有限且工作量大，译文一定还存在许多不足之处。在此敬请各位同行专家学者和广大读者批评指正，欢迎大家将相关意见、建议以及发现的错误发送到邮箱 basics@sjtu.edu.cn。感谢机械工业出版社，尤其是朱劼和唐晓琳编辑，没有她们的信任与支持，整个翻译工作是不可能完成的。

译者

2017.5

自 2010 年年初 Oracle 公司收购 Sun 公司以来, Java 语言迎来激动人心的时代。正如 Java Community Process 计划的活动所印证的, Java 语言将不断发展。本书基于 Java SE 8 (Java Platform Standard Edition 8) 并引用其 API。

相比本书第 5 版(针对的是 Java 版本 7), 第 6 版有一些新特点:

- Lambda 表达式使得功能可作为方法参数使用, 代码作为数据使用。Lambda 表达式描述单个方法接口(也称为功能接口)的实例会更简洁。4.4.8 节介绍该专题。
- 联合使用类型注解和可插拔类型系统可改善类型检测, 重复注解使得同样的注解可应用于声明和类型使用。5.4 节和 5.5 节介绍相关专题。
- 默认方法是接口中实现的方法。使用默认方法, 可将新功能添加到接口中, 并确保与这些接口的老版本代码兼容。6.1.7 节介绍该专题。
- 聚合操作可在元素流上执行功能型操作, 特别是集合上的批量操作, 如串行或并行的 `map-reduce` 转换。12.3 节介绍该专题。
- 在限制恶意程序和 RIA 攻击方面有了一些改进。详情可参考第 16 章、19.5 节和 19.6 节、20.4 节等新增或修订章节。
- 日期/时间 API 可用于表示日期和时间并处理日期和时间值。这些 API 支持国际标准日历系统和其他常见的全球日历。新增的第 21 章介绍该专题。

如果读者准备参加 Java SE 8 认证考试, 本书可提供一定的帮助。附录中列出三个级别的考试, 详细介绍每个级别考试覆盖的内容, 并指出书中涉及相关信息的章节及有价值的参考资料。注意, 这只是参加认证考试要准备的众多内容之一。检查在线教程, 下载最新的认证目标和参考教程的相关章节。

所有材料都经过 Oracle Java 工程组的成员审校, 以确保书中信息都是正确和最新的。本书也被置于 Oracle 网站的在线教程内, 其网址如下:

<http://docs.oracle.com/javase/tutorial>

本书内容被视为 Java 的核心教程, 是初学者和中级程序员的必学知识。只要掌握这些内容, 就可以研究网站上 Java 平台文档的其余部分。如果你有兴趣开发复杂的富互联网应用(RIA), 可以查看 JDK 中的 Java 图形用户界面(GUI)工具包 JavaFX。相关专题参考第 22 章。

和以前的版本一样, 本书的目的是编写一个易于阅读的实用程序员指南, 以协助读者学习使用 Java 提供的丰富环境来构建应用程序、applet 和组件。让我们继续学习编程吧!

读者对象

本书同时面向新手和有经验的程序员。

- 新手通读本书(包括第 1 章中编译和运行第一个程序的每一步指令)收获会很大。
- 有经验的过程式语言(如 C 语言)程序员可以从面向对象的概念和 Java 程序语言的特性开始学习。
- 有经验的程序员可以直接学习高级主题, 如泛型、并发和部署等。

本书内容适用于不同层次的程序员学习。

如何使用本书

本书可直接从头读到尾或忽略某些主题跳到另一个主题。这些信息的组织有一定的逻辑顺序，并尽可能避免引用前文。

本书例子都在 JDK 8 中编译。读者需要下载该版本或更新的版本以编译和运行实例。

本书对引用的一些材料提供了在线帮助（如可下载的实例、问题和习题的答案、JDK 8 手册和 API 规范）。

相关链接在文中以脚注形式给出，例如：

8/docs/api/java/lang/Class.html

和

tutorial/java/generics/examples/BoxDemo.java

Oracle 网站上 Java 文档主页的地址如下：

<http://docs.oracle.com/javase/>

访问脚注中给出的在线文件时，需要将脚注中的 URL 链接在 Java 文档主页之后，例如：

<http://docs.oracle.com/javase/8/docs/api/java/lang/Class.html>

和

<http://docs.oracle.com/javase/tutorial/java/generics/examples/BoxDemo.java>

Java 教程也提供了两种格式的电子书：

- 移动电子书文件，适用于 Kindle。

- ePub 电子书文件，适用于 iPad、Nook 和其他支持 ePub 格式的设备。

每本电子书包含唯一的路径，对应本书的相关章节。读者可通过 Java 教程主页上的链接 “In Book Form” 下载电子书：

<http://docs.oracle.com/javase/tutorial/index.html>

欢迎读者给我们反馈，请通过下述教程反馈页面联系我们：

<http://docs.oracle.com/javase/feedback.html>

致谢

如果没有 Oracle Java 工程组不知疲倦地检查本书的技术内容，本书将不能面世。本书第 6 版的推出，要特别感谢 Alan Bateman、Alex Buckley、Stephen Colebourne、Joe Darcy、Jeff Dinkins、Mike Duigou、Brian Goetz、Andy Herrick、Stuart Marks、Thomas Ng、Roger Riggs、Leif Samuelsson 和 Daniel Smith。

Jordan Douglas 和 Dawn Tyler 快速高效地制作了书中的专业插图。

Janet Blowney、Deborah Owens 和 Susan Shepard 仔细完整地对手稿进行了编辑加工。

感谢团队成员 Devika Gollapudi、Ram Goyal 和 Alexey Zhebel 的支持。

最后（但同样重要）要感谢 Sowmya Kannan、Sophia Mikulinsky、Alan Sommerer 和 Barbara Ramsey 等管理人员的支持。

出版者的话
译者序
前言

第 1 章 快速入门	1
1.1 关于 Java 技术	1
1.1.1 Java 程序语言	1
1.1.2 Java 平台	2
1.1.3 Java 技术的功能	3
1.1.4 Java 技术的优势	3
1.2 “Hello World!” 实例程序	4
1.2.1 用 NetBeans IDE 开发 “Hello World !”	4
1.2.2 在 Microsoft Windows 中开发 “Hello World !”	10
1.2.3 在 Solaris 和 Linux 中开发 “Hello World !”	13
1.3 “Hello World !” 实例程序剖析	15
1.3.1 源码注释	15
1.3.2 HelloWorldApp 类定义	15
1.3.3 main 方法	16
1.4 常见问题 (及其解决方案)	16
1.4.1 编译器问题	16
1.4.2 运行时问题	18
1.5 问题和练习: 快速入门	19
第 2 章 面向对象的编程概念	21
2.1 对象	21
2.2 类	22
2.3 继承	23
2.4 接口	24
2.5 包	25
2.6 问题和练习: 面向对象的编程概念	25

第 3 章 语言基础	27
3.1 变量	27
3.1.1 命名	28
3.1.2 基本数据类型	28
3.1.3 数组	32
3.1.4 小结	35
3.1.5 问题和练习: 变量	36
3.2 运算符	36
3.2.1 赋值运算符、算术运算符和一元运算符	37
3.2.2 等式运算符、关系运算符和条件运算符	40
3.2.3 位运算符和移位运算符	42
3.2.4 小结	42
3.2.5 问题和练习: 运算符	43
3.3 表达式、语句和块	44
3.3.1 表达式	44
3.3.2 语句	45
3.3.3 块	45
3.3.4 问题和练习: 表达式、语句和块	46
3.4 控制流语句	46
3.4.1 if-then 语句和 if-then-else 语句	46
3.4.2 switch 语句	47
3.4.3 while 语句和 do-while 语句	51
3.4.4 for 语句	52
3.4.5 分支语句	53
3.4.6 小结	56
3.4.7 问题和练习: 控制流语句	56
第 4 章 类和对象	58
4.1 类	58
4.1.1 声明类	59

4.1.2 声明成员变量 60

4.1.3 定义方法 61

4.1.4 构建构造器 62

4.1.5 将消息传给方法或构造器 63

4.2 对象 66

4.2.1 创建对象 67

4.2.2 使用对象 69

4.3 类的更多细节 71

4.3.1 从方法返回值 71

4.3.2 使用 `this` 关键字 73

4.3.3 控制对类成员的访问 74

4.3.4 类成员 75

4.3.5 初始化字段 78

4.3.6 小结 79

4.3.7 问题和练习：类 79

4.3.8 问题和练习：对象 80

4.4 嵌套类 81

4.4.1 为什么使用嵌套类 81

4.4.2 静态嵌套类 82

4.4.3 内部类 82

4.4.4 覆盖 82

4.4.5 序列化 83

4.4.6 内部类实例 83

4.4.7 局部类和匿名类 85

4.4.8 `Lambda` 表达式 92

4.4.9 何时使用嵌套类、局部类、匿名类和 `Lambda` 表达式 105

4.4.10 问题和练习：嵌套类 105

4.5 枚举类型 106

4.5.1 问题和练习：枚举类型 108

第 5 章 注解 109

5.1 注解基础知识 109

5.1.1 注解的格式 109

5.1.2 注解的使用场景 110

5.2 声明注解类型 110

5.3 预定义注解类型 111

5.3.1 `Java` 语言使用的注解类型 111

5.3.2 应用于其他注解的注解 112

5.4 类型注解和可插拔类型系统 113

5.5 重复注解 114

5.5.1 声明重复注解类型 114

5.5.2 声明容器注解类型 115

5.5.3 检索注解 115

5.5.4 设计时的注意事项 115

5.6 问题和练习：注解 115

第 6 章 接口与继承 117

6.1 接口 117

6.1.1 `Java` 语言的接口 117

6.1.2 将接口用作 `API` 118

6.1.3 定义接口 118

6.1.4 实现接口 119

6.1.5 将接口用作类型 120

6.1.6 进化接口 121

6.1.7 默认方法 122

6.1.8 小结 129

6.1.9 问题和练习：接口 129

6.2 继承 130

6.2.1 `Java` 平台中类的层次结构 130

6.2.2 继承实例 131

6.2.3 子类能做什么 132

6.2.4 超类的私有成员 132

6.2.5 转换对象 132

6.2.6 状态、实现和类型的多重继承 133

6.2.7 覆盖和屏蔽方法 133

6.2.8 多态性 136

6.2.9 屏蔽字段 138

6.2.10 使用 `super` 关键字 138

6.2.11 将对象用作超类 140

6.2.12 编写 `final` 类和方法 142

6.2.13 抽象方法和类 143

6.2.14 小结 145

6.2.15 问题和练习：继承 145

第 7 章 泛型 147

7.1 为什么用泛型 147

7.2 泛型类型 147

7.2.1 一个简单的 `Box` 类 148

7.2.2	Box 类的泛型版本	148
7.2.3	类型参数命名约定	148
7.2.4	泛型类型的调用和实例化	149
7.2.5	钻石运算符	149
7.2.6	多个类型参数	149
7.2.7	参数化类型	150
7.2.8	原生类型	150
7.3	泛型方法	151
7.4	受限类型形式参数	152
7.4.1	多重限制	153
7.4.2	泛型方法和受限类型形式 参数	154
7.5	泛型、继承和子类型	154
7.5.1	泛型类和子类型	155
7.6	类型推导	156
7.6.1	类型推导和泛型方法	156
7.6.2	类型推导和泛型类的 实例化	157
7.6.3	类型推导与泛型类和非泛型 类的泛型构造函数	157
7.6.4	目标类型	158
7.7	通配符	159
7.7.1	上界通配符	159
7.7.2	无界通配符	160
7.7.3	下界通配符	160
7.7.4	通配符和子类型	161
7.7.5	通配符匹配和辅助方法	162
7.7.6	通配符使用指南	164
7.8	类型擦除	165
7.8.1	泛型类型的擦除	165
7.8.2	泛型方法的擦除	166
7.8.3	类型擦除效果和桥方法	167
7.8.4	不可具体化类型和可变参数 方法	168
7.9	泛型的局限性	170
7.9.1	不能用基本数据类型实例化 泛型类型	170
7.9.2	不能创建类型参数实例	171
7.9.3	不能声明类型为“类型参数” 的静态字段	171
7.9.4	对参数化类型不能用类型转 换或 instanceof 运算符	172
7.9.5	不能创建参数化类型数组	172
7.9.6	不能创建、捕获或抛出参数 化类型的对象	172
7.9.7	每次重载时其形式参数类型 都被擦除为相同的原生类型 的方法不能重载	173
7.10	问题和练习：泛型	173
第 8 章 程序包		175
8.1	程序包的创建和使用	175
8.1.1	程序包的创建	176
8.1.2	程序包的命名	177
8.1.3	程序包成员的使用	177
8.1.4	源文件和类文件的管理	180
8.1.5	小结	182
8.2	问题和练习：创建和使用包	182
第 9 章 数字和字符串		183
9.1	数字	183
9.1.1	Number 类	183
9.1.2	格式化数字打印输出	184
9.1.3	其他数学运算方法	188
9.1.4	自动装箱和拆箱	191
9.1.5	小结	193
9.1.6	问题和练习：数字	193
9.2	字符	194
9.2.1	转义字符	194
9.3	字符串	195
9.3.1	创建字符串	195
9.3.2	字符串长度	196
9.3.3	字符串连接	196
9.3.4	创建格式字符串	197
9.3.5	数字和字符串之间的转换	197
9.3.6	操作字符串中的字符	199
9.3.7	比较字符串和字符串的 子串	202
9.3.8	StringBuilder 类	203
9.3.9	小结	207

9.3.10 问题和练习：字符和字符串	207	11.1.4 扫描和格式化	234
第 10 章 异常	209	11.1.5 命令行 I/O	238
10.1 什么是异常	209	11.1.6 数据流	240
10.2 捕获或指明规定	210	11.1.7 对象流	241
10.2.1 三类异常	210	11.2 文件 I/O (以 NIO.2 为特征)	243
10.2.2 绕过捕获或指明	210	11.2.1 什么是路径 (以及其他文件 系统情况)	243
10.3 捕获和处理异常	211	11.2.2 Path 类	245
10.3.1 try 块	211	11.2.3 文件操作	250
10.3.2 catch 块	212	11.2.4 检查文件或目录	253
10.3.3 finally 块	213	11.2.5 删除文件或目录	254
10.3.4 try-with-resources 语句	214	11.2.6 复制文件或目录	254
10.3.5 汇总	216	11.2.7 移动文件或目录	255
10.4 指明一个方法抛出的异常	218	11.2.8 管理元数据 (文件和文件 存储属性)	255
10.5 如何抛出异常	219	11.2.9 读取、写入和创建文件	261
10.5.1 throw 语句	219	11.2.10 随机访问文件	266
10.5.2 Throwable 类及其 子类	220	11.2.11 创建和读取目录	267
10.5.3 Error 类	220	11.2.12 符号链接或其他方式的 链接	270
10.5.4 Exception 类	220	11.2.13 遍历文件树	272
10.5.5 链式异常	220	11.2.14 查找文件	275
10.5.6 创建异常类	221	11.2.15 监视目录的变化	278
10.6 未检查异常：争议	222	11.2.16 其他有用的方法	283
10.7 异常的优点	223	11.2.17 遗留文件的 I/O 代码	284
10.7.1 优点 1：把错误处理代码和 “正规”代码分离开	223	11.3 小结	286
10.7.2 优点 2：根据调用栈上传 错误	224	11.4 问题和练习：基本 I/O	286
10.7.3 优点 3：对错误类型进行 分组并加以区分	226	第 12 章 集合	287
10.8 小结	226	12.1 集合简介	287
10.9 问题和练习：异常	227	12.1.1 集合框架是什么	287
第 11 章 基本 I/O 和 NIO.2	229	12.1.2 Java 集合框架的好处	288
11.1 I/O 流	229	12.2 接口	288
11.1.1 字节流	230	12.2.1 Collection 接口	290
11.1.2 字符流	231	12.2.2 遍历集合	291
11.1.3 缓冲流	233	12.2.3 Collection 接口的批量 操作	292
		12.2.4 Collection 接口的数组 操作	293
		12.2.5 Set 接口	293

12.2.6	List 接口	296	第 13 章 并发	351
12.2.7	Queue 接口	302	13.1 进程和线程	351
12.2.8	Deque 接口	303	13.1.1 进程	351
12.2.9	Map 接口	304	13.1.2 线程	351
12.2.10	对象排序	310	13.2 线程对象	352
12.2.11	SortedSet 接口	315	13.2.1 定义和启动一个线程	352
12.2.12	SortedMap 接口	317	13.2.2 使用 sleep 方法暂停 执行	353
12.2.13	小结	318	13.2.3 中断	353
12.2.14	问题和练习: 接口	318	13.2.4 联合	354
12.3	聚合操作	319	13.2.5 SimpleThreads 实例	355
12.3.1	管道和流	320	13.3 同步	356
12.3.2	聚合操作和迭代器之间的 差异	321	13.3.1 线程冲突	356
12.3.3	归约	321	13.3.2 内存一致性错误	357
12.3.4	并行	325	13.3.3 同步方法	358
12.3.5	副作用	328	13.3.4 内部锁和同步	359
12.3.6	问题和练习: 聚合操作	330	13.3.5 原子访问	360
12.4	实现	331	13.4 活性	361
12.4.1	Set 实现	333	13.4.1 死锁	361
12.4.2	List 实现	334	13.4.2 饥饿和活锁	362
12.4.3	Map 实现	335	13.5 保护块	362
12.4.4	Queue 实现	336	13.6 不可变对象	365
12.4.5	Deque 实现	337	13.6.1 同步类实例	365
12.4.6	封装实现	338	13.6.2 定义不可变对象的策略	367
12.4.7	简单实现	340	13.7 高级并发对象	368
12.4.8	小结	341	13.7.1 锁对象	368
12.4.9	问题和练习: 实现	341	13.7.2 执行器	370
12.5	算法	342	13.7.3 并发集合	374
12.5.1	排序	342	13.7.4 原子变量	375
12.5.2	混排	344	13.7.5 并发随机数	376
12.5.3	常规数据操作	344	13.8 问题和练习: 并发	376
12.5.4	查询	344		
12.5.5	组合	345	第 14 章 正则表达式	378
12.5.6	查找极值	345	14.1 简介	378
12.6	自定义集合实现	345	14.1.1 正则表达式	378
12.6.1	编写实现的原因	345	14.1.2 正则表达式的表示方法	378
12.6.2	如何编写自定义实现	346	14.2 测试工具	379
12.7	互操作性	347	14.3 字符串文字	380
12.7.1	兼容性	347	14.3.1 元字符	380
12.7.2	API 设计	349	14.4 字符类	381

14.4.1 简单类	381	14.11 PatternSyntaxException 类方法	401
14.5 预定义字符类	384	14.12 Unicode 支持	403
14.6 量词	386	14.12.1 匹配特定代码点	403
14.6.1 零长度匹配	387	14.12.2 Unicode 字符属性	403
14.6.2 捕捉组和带量词的字符类	389	14.13 问题和练习: 正则表达式	404
14.6.3 贪婪型、勉强型和占有型 量词之间的区别	390		
14.7 捕捉组	390	第 15 章 平台环境	405
14.7.1 编号	390	15.1 配置工具	405
14.7.2 反向引用	391	15.1.1 属性	405
14.8 边界匹配器	391	15.1.2 命令行参数	408
14.9 Pattern 类方法	393	15.1.3 环境变量	409
14.9.1 使用标记创建模式	393	15.1.4 其他配置工具	410
14.9.2 嵌套标记表达式	394	15.2 系统工具	410
14.9.3 使用 matches(String, CharSequence) 方法	395	15.2.1 命令行 I/O 对象	410
14.9.4 使用 split(String) 方法	395	15.2.2 系统属性	410
14.9.5 其他实用方法	396	15.2.3 安全管理器	413
14.9.6 java.lang.String 中 Pattern 方法的等价 方法	396	15.2.4 系统的其他方法	414
14.10 Matcher 类方法	397	15.3 PATH 和 CLASSPATH 环境 变量	414
14.10.1 索引方法	397	15.3.1 更新 PATH 环境变量 (Microsoft Windows)	414
14.10.2 学习方法	397	15.3.2 更新 PATH 环境变量 (Solaris、 Linux 和 OS X)	416
14.10.3 替换方法	397	15.3.3 检查 CLASSPATH 环境变量 (所有平台)	416
14.10.4 使用 start 方法和 end 方法	398	15.4 问题和练习: 平台环境	417
14.10.5 使用 matches 方法和 lookingAt 方法	399	第 16 章 JAR 文件	418
14.10.6 使用 replaceFirst(Str- ing) 方法和 replace- All(String) 方法	399	16.1 JAR 文件使用入门	418
14.10.7 使用 appendReplacement (StringBuffer, String) 方法和 appendTail(Str- ingBuffer) 方法	400	16.1.1 创建 JAR 文件	419
14.10.8 java.lang.String 中 Matcher 方法的等价方法	401	16.1.2 查看 JAR 文件内容	421
		16.1.3 抽取 JAR 文件内容	422
		16.1.4 更新 JAR 文件	423
		16.1.5 运行打包为 JAR 的 软件	424
		16.2 清单文件使用入门	426
		16.2.1 理解默认的清单文件	426
		16.2.2 修改清单文件	426

16.2.3 设置应用程序的入口点	427	17.5 Java Web Start 与安全	448
16.2.4 将类文件加入 JAR 文件的 类路径	428	17.5.1 动态下载 HTTPS 认证	449
16.2.5 设置包版本信息	429	17.6 Java Web Start 常见问题	449
16.2.6 用 JAR 文件封装包	430	17.7 问题和练习: Java Web Start	450
16.2.7 使用清单属性增强 安全性	430	第 18 章 applet	451
16.3 JAR 文件的签名和验证	431	18.1 开始使用 applet	451
16.3.1 理解签名和验证	431	18.1.1 定义 Applet 类的子类	452
16.3.2 对 JAR 文件签名	433	18.1.2 里程碑方法	452
16.3.3 验证签名的 JAR 文件	435	18.1.3 applet 的生命周期	452
16.4 使用 JAR 相关 API	435	18.1.4 applet 的执行环境	454
16.4.1 实例: JarRunner 应用	436	18.1.5 开发 applet	454
16.4.2 JarClassLoader 类	436	18.1.6 部署 applet	457
16.4.3 JarRunner 类	438	18.2 applet 的更多功能	459
16.5 问题和练习: JAR 文件	439	18.2.1 查找和加载数据文件	459
第 17 章 Java Web Start	440	18.2.2 定义和使用 applet 参数	460
17.1 开发 Java Web Start 应用	440	18.2.3 显示简短的状态字符串	462
17.1.1 创建顶层 JPanel 类	441	18.2.4 在浏览器中显示文档	462
17.1.2 创建应用	441	18.2.5 从 applet 调用 JavaScript 代码	464
17.1.3 从最后部署机制中分离出 核心方法的好处	442	18.2.6 从 JavaScript 代码中调用 applet 方法	465
17.1.4 获取资源	442	18.2.7 通过事件句柄处理初始化 状态	468
17.2 部署 Java Web Start 应用	442	18.2.8 操纵 applet 网页的 DOM	469
17.2.1 设置 Web 服务器	445	18.2.9 显示自定义的加载进度指 示器	471
17.3 显示自定义的加载进度 指示器	445	18.2.10 将诊断写入标准输出和错 误流	474
17.3.1 开发自定义的加载进度指 示器	445	18.2.11 开发可拖动的 applet	475
17.3.2 为 Java Web Start 应用指定 自定义的加载进度指示器	447	18.2.12 和其他 applet 交互	477
17.4 运行 Java Web Start 应用	447	18.2.13 与服务器端应用交互	478
17.4.1 通过浏览器运行 Java Web Start 应用	448	18.2.14 applet 能做什么和不能做 什么	480
17.4.2 通过 Java Cache Viewer 运行 Java Web Start 应用	448	18.3 applet 常见问题及解决方案	481
17.4.3 从桌面运行 Java Web Start 应用	448	18.4 问题和练习: applet	482
		第 19 章 Java 富互联网应用系统	483
		19.1 设置可信参数和安全属性	483

19.1.1 系统属性	484	21.2 日期 / 时间 API 设计原则	510
19.2 JNLP API	485	21.2.1 明确性	510
19.2.1 通过 JNLP API 访问客 户端	486	21.2.2 流式	510
19.3 cookie	489	21.2.3 不可变性	511
19.3.1 cookie 的类型	489	21.2.4 可扩展性	511
19.3.2 RIA 中的 cookie 支持	489	21.3 日期 / 时间程序包	511
19.3.3 获取 cookie	489	21.4 方法命名约定	511
19.4 自定义加载体验	491	21.5 标准日历	512
19.5 RIA 的安全	491	21.6 日期 / 时间类概述	512
19.6 安全 RIA 手册	492	21.7 DayOfWeek 和 Month 枚举器	513
19.6.1 遵循安全编程指导	492	21.7.1 DayOfWeek 枚举器	513
19.6.2 使用最新版 JRE 测试	492	21.7.2 Month 枚举器	513
19.6.3 加入清单属性	492	21.8 日期类	515
19.6.4 使用已签署的 JNLP 文件	492	21.8.1 LocalDate	515
19.6.5 签署 JAR 文件并加盖时 间戳	493	21.8.2 YearMonth	515
19.6.6 使用 HTTPS 协议	493	21.8.3 MonthDay	516
19.6.7 避免本地 RIA	493	21.8.4 Year	516
19.7 问题和练习: Java 富互联网应 用系统	493	21.9 日期和时间类	516
21.9.1 LocalDateTime	516	21.9.2 LocalDateTime	517
第 20 章 深入理解部署	495	21.10 时区和时区偏移类	517
20.1 RIA 的用户接纳	495	21.10.1 ZoneId 和 ZoneOff- set	517
20.2 部署工具	496	21.10.2 日期 / 时间类	518
20.2.1 部署工具脚本所在位置	496	21.11 Instant 类	520
20.2.2 部署 applet	496	21.12 解析和格式化	521
20.2.3 部署 Java Web Start 应用	499	21.12.1 解析	522
20.2.4 检查客户端的 JRE 软件 版本	501	21.12.2 格式化	522
20.3 Java 网络加载协议	501	21.13 时间程序包	523
20.3.1 JNLP 文件结构	501	21.13.1 Temporal 和 Temporal- Accessor	523
20.4 部署的最佳实践	505	21.13.2 ChronoField 和 IsoFields	524
20.4.1 减少下载时间	505	21.13.3 ChronoUnit	524
20.4.2 避免不必要的更新检查	506	21.13.4 时间调节器	524
20.4.3 确保 JRE 软件存在	508	21.13.5 时间查询	526
20.5 问题和练习: 深入理解部署	509	21.14 周期和持续时间	527
第 21 章 日期 / 时间 API	510	21.14.1 Duration	528
21.1 日期 / 时间 API 概述	510	21.14.2 ChronoUnit	528
		21.14.3 Period	528

21.15 时钟	529	21.17.3 日期和时间的格式化	534
21.16 非 ISO 日期的转换	530	21.18 小结	534
21.16.1 转换成非 ISO 日期	530	21.19 问题和练习：日期 / 时间	
21.16.2 转换成 ISO 日期	531	API	535
21.17 遗留的日期 / 时间代码	532	第 22 章 JavaFX 简介	536
21.17.1 与遗留代码的互操作	532	附录 Java 程序语言认证考试复习	
21.17.2 将 <code>java.util</code> 日期和时间		大纲	537
功能映射给 <code>java.time</code>	533		

快速入门

本章介绍 Java 程序语言的入门知识。1.1 节概述 Java 技术，通过说明 Java 技术的功能及其如何简化编程讨论 Java 程序语言及其平台。1.2 节引入“Hello World!”实例程序，主要描述创建该程序需要下载、安装和输入的内容，并说明在 NetBeans IDE、Microsoft Windows、Solaris、Linux 和 OS X 等平台上创建该程序的完整过程。1.3 节详细讨论“Hello World!”实例程序的代码，包括源码注释、HelloWorldApp 类定义块和 main 方法。1.4 节介绍一些常见问题及其解决方案，编译或运行程序出现问题时可查阅本节内容。最后列出一些问题和练习供读者自行测试。

1.1 关于 Java 技术

大家都在谈 Java 技术，但 Java 技术到底是什么？本节解释 Java 技术如何成为程序语言和平台，并概述 Java 技术的功能特性。

1.1.1 Java 程序语言

Java 程序语言是一种高级的编程语言，它具备如下性质：

- 简单
- 面向对象
- 分布式
- 多线程
- 动态
- 架构中立
- 可移植
- 高性能
- 强壮
- 安全

上述术语的定义可参考 James Gosling 和 Henry McGilton 的白皮书《The Java Language Environment》^①。

图 1-1 描述了 Java 应用程序开发的完整过程。首先将所有源代码都写在扩展名为 .java 的纯文本文件中，接着通过编译器（javac）将那些源文件编译成 .class 文件。.class 文件里的代码不是本地代码，而是字节码（bytecode）——Java 虚拟机（Java Virtual Machine, JVM）^②能识别的机器语言。最后通过启动器（java）在 JVM 实例中运行应用程序。

① <http://www.oracle.com/technetwork/java/langenv-140151.html>

② 术语 Java Virtual Machine 和 Java VM 都表示 Java 平台的虚拟机器，即 Java 虚拟机。