

Broadview®
www.broadview.com.cn

Pearson

写给大忙人的 *Java SE 9* 核心技术

[美] Cay S. Horstmann 著

杨谦 章历 谢志强 译

Core Java SE 9
for the Impatient
Second Edition



中国工信出版集团

电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

写给大忙人的 *Java SE 9*核心技术

[美] Cay S. Horstmann 著

杨谦 章历 谢志强 译

Core Java SE 9
for the Impatient
Second Edition

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内容简介

本书是经典著作《Java核心技术》(Core Java)作者Cay S. Horstmann的全新力作,书中以关键的核心技术为着眼点,以全局高度提炼语言精髓,以高浓度、高含金量的方式对Java SE 9的新特性进行细致讲解。作者将其沉浸于Java多年的真知灼见蕴藏于书中,既扎实覆盖Java编程的基础知识,又独具匠心地从Java 9及其重要的新特性之一——模块发端,带领读者一路探索和实践Java的诸多新特性,诸如JShell、集合类工厂方法、改进的Stream API、私有接口、HTTP/2、并发编程中的增强部分、全新的时间/日期API,以及脚本语言在Java中的使用。

“Impatient”系列图书风格鲜明,除悉数出自大师之手外,更以文笔轻松、易读易懂、富含生动且可用于实践的示例程序(下载源代码后可直接运行)而见长。这种亲身实践的讲授模式,非常便于读者加深对技术的理解和掌握。初学者可用来参考入门;有开发经验者也可用来巩固编程核心知识,与时俱进地把握技术发展现状与未来趋势。

本书适合有一定Java开发经验的工程师或者架构师阅读,也可以作为Java新特性的参考手册。

Authorized translation from the English language edition, entitled Core Java SE 9 for the Impatient, Second Edition, ISBN: 978-0-13-469472-6 by Cay S. Horstmann, published by Pearson Education, Inc, Copyright © 2018 Pearson Education, Inc.

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 PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, Copyright © 2018

本书简体中文版专有版权由Pearson Education培生教育出版集团授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书简体中文版贴有Pearson Education培生教育出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2017-7534

图书在版编目(CIP)数据

写给大忙人的Java SE 9核心技术 / (美)凯·S.霍斯特曼(Cay S. Horstmann)著;杨谦等译. —北京:电子工业出版社, 2018.10

书名原文: Core Java SE 9 for the Impatient, Second Edition

ISBN 978-7-121-34278-3

I. ①写… II. ①凯… ②杨… III. ①JAVA语言—程序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第107777号

策划编辑:张春雨

责任编辑:李云静

印刷:三河市君旺印务有限公司

装订:三河市君旺印务有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱

邮编:100036

开本:787×980 1/16 印张:34.5 字数:613千字

版次:2018年10月第1版

印次:2018年10月第1次印刷

定价:128.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至zltz@phei.com.cn, 盗版侵权举报请发邮件至dbqq@phei.com.cn。

本书咨询联系方式:010-51260888-819, faq@phei.com.cn。

推荐序一

1991年，来自Sun公司的一个名为“Green Team”的小团队的3个人：James Gosling、Mike Sheridan和Patrick Naughton发布了一个叫作“Greentalk”的语言，其最初设计目标是在小型嵌入式设备中运行应用程序，这也就是Java语言的雏形。自1995年发行的第一个Alpha版本开始，Java距今已有23年历史了，可谓经久不衰，算得上历史上最成功的语言了。

2017年9月21日，Java 9发布了，引入了诸多新特性，例如，集合类工厂方法、私有接口、支持HTTP/2，以及并发处理的改进等。其实做Java Web开发的时间越长，越需要对Java底层实现深入了解。通过本书，能够准确理解Java技术的走向，帮助Java开发者从JDK层面重构应用的代码。另外，本书是国内目前关于Java 9为数不多的中文版书籍，而本书的译者是我在IBM多年的同事，其一直专注于Java开发。本书的翻译质量较高，通俗易懂，用词准确，通过阅读书中内容，能够帮助广大Java开发者更快、更准确地了解Java 9的特性。

王青

杰蛙科技（北京）有限公司 首席架构师

推荐序二

我曾经在IBM一起战斗过的兄弟杨谦邀请我为本书作序，我虽然心怀忐忑，但还是欣然应允了。

Java老手都会知道*Core Java*（《Java核心技术》）这部大作，本书的作者Cay S. Horstmann也正是*Core Java*的主要作者。他是圣荷西州立大学计算机系的教授、Java语言的积极倡导者，他能够以形象的语言，向大家阐述抽象的Java概念，让初学者能够很容易地掌握Java这门语言。然而光阴似箭，Java到今天已经经历了二十多个寒暑。这么长的时间里，Java也已经演化到了9.0版本，许多更先进的思想和更简洁的API被加入进来，同时也抛弃了很多过时的内容。在这样的情况下，*Core Java*这本大部头就多多少少显得有些不合时宜了，这也是Cay S. Horstmann写作本书的原因。在本书中，Cay S. Horstmann针对Java应用开发人员，提炼出了Java最核心、最新鲜、工作中最常用到的部分，舍弃了诸如Swing等在现实工作中很少用到的内容，让读者在最短的时间里尽情吸收Java语言的精华，以快速地把所学的知识应用到实际工作中去。

本书的译者之一杨谦是我在IBM CDL时朝夕相处的同事，他有着十多年的Java实战经验和深厚的技术功底，同时还秉承了IBM一贯的科学严谨作风，再加上长期外企工作环境的熏陶，锻炼出了对英文技术资料很好的阅读理解能力。我相信，他们完全能够将本书中的内容用中文清晰、准确地呈献给读者。

霍振中

北京大米未来科技有限公司联合创始人、CTO

译者序

2017年Oracle发布了Java 9。Java 9引入了大家期待已久的模块功能以及很多其他特性，例如，JShell、集合类工厂方法、改进的Stream API、私有接口、HTTP/2等。本书特色鲜明，它不仅为开发者全面介绍这些新特性，而且从Java的最核心基础——类、继承、接口、反射、泛型编程、集合等开始。全书的开始几章全面细致、重点突出地介绍Java的核心基础并在第3章穿插了lambda表达式。之后各章全面介绍Java的新特性和原有功能的改进，其中第8章详细介绍Stream的使用，第10章介绍最新的并发编程API，第15章详细介绍模块。全书内容翔实，实例丰富，理论与实践并举。

译者在翻译本书时，一边感到万分的荣幸，一边又感到责任重大。吉尔伯特·海特（美国教育家）曾经说过：写了一本很糟糕的书只是犯错而已，而把一本好书翻译得很糟糕则是犯罪。但是对于本书这样的Java核心技术和最新特性之作，即便是冒着犯罪的风险，也值得翻译出来并推荐给大家。

本书承载的内容丰富而深入。在翻译的过程中，译者也常感汗颜，原来自诩对Java新特性颇为了解的我深刻感受到自己知识面的浅薄和不完整，于是翻译过程也成为自己检讨和学习的过程，收获颇丰。本书的作者是经典书籍*Core Java*（《Java核心技术》）的主要作者，他有着丰富的Java知识，并一直在普及介绍Java的最新特性。因此，希望大家反复阅读本书，如此一来必定受益无穷。

由于本书作者的知识广博、经验丰富，而译者本人知识有限、时间仓促，难免使得原文的个别之处无法原汁原味地呈现给读者，在此敬请各位读者多多包涵，不吝指教，并帮忙指出纰漏。我也希望能就本书的内容与读者有更多的交流，联系邮箱是yqbjtu@163.com。

最后，我要感谢电子工业出版社博文视点公司的张春雨编辑在译书的过程中给予我们的莫大帮助。我还要感谢本书的其他译者章历、王巍、高伟斌、张媛、田敏、王

雪芳、杨建轩、康引鸽、刘增涛、杨燕、史肖玲、张会平、郝明、刘香兰、田丰产、刘回良和东莞经济贸易学校的谢志强老师等。在此还要特别感谢杰蛙科技（北京）有限公司的首席架构师王青先生和北京大米未来科技有限公司的CTO霍振中先生，他们的参与使得本书的质量得到进一步保证。

杨 谦

2018年9月于北京志新村

谨以此书献给Chi，那个对我最有耐心的人。

前 言

Java已诞生20多年，经典书籍*Core Java*（《Java核心技术》）卷I和卷II用2000多页全面细致地介绍了Java语言和所有核心类库，以及各版本之间的大量变化。但是如果你想更高效地使用现代Java，那么本书提供了一种更快、更方便地学习Java语言和核心类库的途径。在本书中，我们不会追溯历史一直介绍过去版本的特性。我们将通过Java 9，展示现代Java中的“优秀之处”，因此你可以快速地将它们应用到工作中。

与我之前的“*Impatient*”系列书籍类似，本书也会开门见山地告诉你解决编程问题所需要知道的内容，但不会就某个范例比另外一个范例更有优势而展开讲解。我们会采用有序且篇幅较小的方式展示信息，这样需要信息时，你就可以快速检索获取。

假设你精通其他编程语言，例如C++、JavaScript、Objective-C、PHP或Ruby，通过本书，你将学习如何成为一个称职的Java程序员。我们会涵盖Java开发者需要知道的方方面面，包括功能强大的lambda表达式和Stream。我会告诉你哪里可以找到关于旧式概念的更多信息，你可能依然会在一些遗留代码中看到这些旧式概念，但是我们不会就此展开。

使用Java的一个关键原因是解决并发编程问题。使用Java类库中现成的并行算法和线程安全的数据结构，彻底改变了程序员处理并发编程的方式。我们会提供新的示例，告诉你如何使用强大的类库特性代替那些易出错、低水平的数据结构。

传统的Java书籍关注UI编程——但是现在，很少有开发者在桌面计算机开发用户界面了。如果你想将Java用在服务器端编程或者Android编程，就可以更有效地使用本书，而不被桌面GUI代码干扰了。

最后，本书是写给应用软件编程者的，不是大学教科书，也不适合作为系统向导。本书包含了一些应用软件编程者需要设法解决的问题，例如日志和文件使用，但是你不必学习如何实现链表或者开发一个Web服务器。

希望你喜欢对现代Java的快速介绍，同时希望它能提高你的Java开发工作效率并愉快地使用Java。

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- **下载资源：**本书提供示例代码及资源文件，可在 **下载资源** 处下载。
- **提交勘误：**您对书中内容的修改意见可在 **提交勘误** 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- **交流互动：**在页面下方 **读者评论** 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/34278>



致 谢

一如既往地感谢我的编辑Greg Doech，是他热情地支持我写作一本简短图书，介绍Java SE 9。Dmitry Kirsanov和Alina Kirsanova再次以神奇的速度，细致地将XHTML手稿转变成引人入胜的图书。特别感谢优秀的评审团队，他们发现了很多错误并给出了深思熟虑的改进建议。他们是Andres Almiray、Gail Anderson、Paul Anderson、Marcus Biel、Brian Goetz、Marty Hall、Mark Lawrence、Doug Lea、Simon Ritter、Yoshiki Shibata和Christian Ullenboom。

Cay S. Horstmann

旧金山

2017年7月

作者简介

Cay S. Horstmann是*Java SE 8 for the Really Impatient*和*Scala for the Impatient*（都是Addison-Wesley出版社的书）两本书的作者，也是*Core Java™, Volumes I and II, Tenth Edition*（Prentice Hall于2016年出版）一书的主要作者。他还编写了一系列针对专业编程人员和计算机专业学生的书籍。他是圣荷西州立大学计算机系的教授，也是一位Java Champion¹。

¹ Java Champion是Oracle公司针对Java软件开发社区中有突出贡献的个人授予的一种荣誉。——译者注

目 录

第1章 基本的编程结构	1
1.1 第一个程序	2
1.1.1 "Hello,World!"程序解析.....	2
1.1.2 编译与运行Java程序.....	3
1.1.3 方法调用	5
1.1.4 JShell	7
1.2 基本类型	11
1.2.1 有符号整型.....	11
1.2.2 浮点类型	12
1.2.3 char型	13
1.2.4 布尔型.....	14
1.3 变量.....	14
1.3.1 变量的声明.....	14
1.3.2 名称	14
1.3.3 初始化.....	15
1.3.4 常量.....	15
1.4 算术操作	17
1.4.1 赋值.....	18
1.4.2 基本运算	18
1.4.3 数学方法	20
1.4.4 数字类型转换	21
1.4.5 关系操作符和逻辑操作符	22
1.4.6 大数	24
1.5 字符串	24
1.5.1 字符串的连接	25
1.5.2 子字符串	26

1.5.3	字符串比较	26
1.5.4	数字与字符串转换	28
1.5.5	String类API	29
1.5.6	编码点和编码单元	31
1.6	输入与输出	33
1.6.1	读取输入	33
1.6.2	格式化输出	35
1.7	控制流	37
1.7.1	分支	37
1.7.2	循环	39
1.7.3	跳出循环与继续循环	41
1.7.4	局部变量作用域	43
1.8	数组和数组列表	44
1.8.1	使用数组	44
1.8.2	构造数组	45
1.8.3	数组列表	47
1.8.4	基本类型包装类	48
1.8.5	增强的for循环	48
1.8.6	数组与数组列表的复制	49
1.8.7	数组算法	50
1.8.8	命令行参数	51
1.8.9	多维数组	52
1.9	功能分解	55
1.9.1	静态方法的声明与调用	55
1.9.2	数组参数与返回值	55
1.9.3	可变参数	56
	练习	57
第2章	面向对象编程	61
2.1	使用对象	62
2.1.1	访问器方法和修改器方法	64
2.1.2	对象引用	65
2.2	实现类	67
2.2.1	实例变量	67
2.2.2	方法头	67

2.2.3	方法体	68
2.2.4	实例方法调用	69
2.2.5	this引用	69
2.2.6	值调用	70
2.3	构造对象	72
2.3.1	实现构造函数	72
2.3.2	重载	73
2.3.3	调用另一个构造函数	74
2.3.4	默认初始化	74
2.3.5	实例变量的初始化	75
2.3.6	final实例变量	76
2.3.7	无参构造函数	76
2.4	静态变量和方法	77
2.4.1	静态变量	77
2.4.2	静态常量	78
2.4.3	静态初始块	79
2.4.4	静态方法	80
2.4.5	工厂方法	81
2.5	包	82
2.5.1	包的声明	82
2.5.2	jar命令	83
2.5.3	类路径	84
2.5.4	包访问权限	86
2.5.5	导入类	87
2.5.6	静态导入	88
2.6	嵌套类	89
2.6.1	静态嵌套类	89
2.6.2	内部类	91
2.6.3	内部类的特殊语法	93
2.7	文档注释	95
2.7.1	插入注释	95
2.7.2	类注释	96
2.7.3	方法注释	97
2.7.4	变量注释	97
2.7.5	通用注释	98

2.7.6	链接	98
2.7.7	包、模块和概述注释	99
2.7.8	注释的提取	99
	练习	100
第3章	接口和lambda表达式	105
3.1	接口	106
3.1.1	声明接口	106
3.1.2	实现接口	107
3.1.3	转换为接口类型	109
3.1.4	强制类型转换和instanceof操作符	109
3.1.5	继承接口	110
3.1.6	实现多个接口	111
3.1.7	常量	111
3.2	静态方法、默认方法和私有方法	112
3.2.1	静态方法	112
3.2.2	默认方法	112
3.2.3	解决默认方法冲突的问题	113
3.2.4	私有方法	115
3.3	接口示例	115
3.3.1	Comparable接口	115
3.3.2	Comparator接口	117
3.3.3	Runnable接口	118
3.3.4	UI回调	119
3.4	lambda表达式	120
3.4.1	lambda表达式语法	121
3.4.2	函数式接口	122
3.5	方法引用和构造函数引用	123
3.5.1	方法引用	123
3.5.2	构造函数引用	125
3.6	使用lambda表达式	126
3.6.1	实现延迟执行	126
3.6.2	选择函数式接口	127
3.6.3	实现自己的函数式接口	129
3.7	lambda表达式和变量作用域	130

3.7.1	lambda表达式的作用域	130
3.7.2	访问来自闭合作用域的变量	131
3.8	高阶函数	133
3.8.1	返回函数的方法	134
3.8.2	修改函数的方法	134
3.8.3	Comparator方法	135
3.9	局部类和匿名类	136
3.9.1	局部类	136
3.9.2	匿名类	137
	练习	138
第4章	继承与反射	143
4.1	继承一个类	144
4.1.1	父类与子类	144
4.1.2	定义和继承子类方法	145
4.1.3	方法覆盖	145
4.1.4	子类的构造	147
4.1.5	父类赋值	147
4.1.6	转换	148
4.1.7	final方法和类	149
4.1.8	抽象方法和类	149
4.1.9	受保护访问	150
4.1.10	匿名子类	151
4.1.11	继承和默认方法	152
4.1.12	带super的方法表达式	153
4.2	Object: 终极父类	153
4.2.1	toString方法	154
4.2.2	equals方法	156
4.2.3	hashCode方法	159
4.2.4	克隆对象	160
4.3	枚举	163
4.3.1	枚举方法	164
4.3.2	构造函数、方法和域	165
4.3.3	实例的实现体	165
4.3.4	静态成员	166