

O'REILLY®

TURING

图灵程序设计丛书



# Java攻略

Java常见问题的简单解法

Modern Java Recipes

70余个提炼自开发人员日常工作的范例，涵盖Java 8和Java 9新特性

[美] 肯·寇森 著  
蒋楠 译

 中国工信出版集团

 人民邮电出版社  
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

# Java攻略

## Java常见问题的简单解法

Modern Java Recipes:  
Simple Solutions to Difficult Problems in Java 8 and 9

[美] 肯·寇森 著  
蒋楠 译

Beijing • Boston • Farnham • Sebastopol • Tokyo



O'Reilly Media, Inc. 授权人民邮电出版社出版

人民邮电出版社  
北 京

## 图书在版编目 ( C I P ) 数据

Java攻略：Java常见问题的简单解法 / (美) 肯·寇森 (Ken Kousen) 著；蒋楠译. — 北京：人民邮电出版社，2018. 8

(图灵程序设计丛书)

ISBN 978-7-115-48880-0

I. ①J… II. ①肯… ②蒋… 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155420号

## 内 容 提 要

本书旨在让读者迅速掌握 Java 8 和 Java 9 相关特性，并给出了 70 余个可以用于实际开发的示例，介绍了如何利用这些新特性解决这些问题，从而以更自然的方式让开发人员掌握 Java。

本书适合 Java 开发人员阅读。

---

◆ 著 [美] 肯·寇森  
译 蒋楠  
责任编辑 朱巍  
责任印制 周昇亮

◆ 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>  
三河市君旺印务有限公司印刷

◆ 开本：800×1000 1/16  
印张：15.5  
字数：366千字 2018年8月第1版  
印数：1-3 000册 2018年8月河北第1次印刷  
著作权合同登记号 图字：01-2018-3407号

---

定价：69.00元

读者服务热线：(010)51095186转600 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

站在巨人的肩上  
**Standing on Shoulders of Giants**



iTuring.cn

站在巨人的肩上  
**Standing on Shoulders of Giants**



[iTuring.cn](http://iTuring.cn)

---

# 版权声明

© 2017 by Ken Kousen.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and Posts & Telecom Press, 2018. Authorized translation of the English edition, 2017 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版，2017。

简体中文版由人民邮电出版社出版，2018。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有，未得书面许可，本书的任何部分和全部不得以任何形式重制。

---

# O'Reilly Media, Inc.介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 *Make* 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过图书出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

## 业界评论

“O'Reilly Radar 博客有口皆碑。”

——*Wired*

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——*Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——*Linux Journal*

Xanden, 这本书是送你的礼物, 想不到吧!

---

# 译者序

与其他语言相比，Java 最大的优势或许在于完善的生态系统：开发者所需要的一切，几乎都能从这个生态系统中找到。世界上累计有 150 亿台设备运行 Java，全球 Java 开发者的数量超过 1000 万人，Java 不仅构成了大量开源平台的基础，也已成为软件文化中不可或缺的一部分。

然而，作为一门诞生于 1995 年的语言，运行环境臃肿、代码库庞大等问题逐渐成为制约 Java 发展的瓶颈。对于稳定性与兼容性的顾虑，使得这门语言越来越难以大刀阔斧地改革。相当一部分业务服务仍然采用 Java 8 之前的版本构建，复杂的系统升级往往令维护人员望而却步。并非企业不想求变，而是求变的代价在某些时候显得异常高昂。

但 Oracle 从未停止探索的脚步，Java 9 的发布或许可以视作 Java 平台求变的开始。尽管社区对 Jigsaw 项目褒贬不一，模块化系统的意义仍然有待时间检验，不过 Oracle 意欲求变的决心由此可见一斑。而在 Java 9 面世之后，Oracle 加快了这门语言的迭代速度，版本发布周期改为半年一次，以便缩短开发者使用新功能的时间。Java 的发布速度经常受到诟病，这种改变或许有助于解决这个问题。

硬件厂商同样在探索前行。作为 JavaOne 2017 赞助商之一的英特尔在向量计算领域投入大量精力，发布了有助于 Java API 充分利用硬件向量计算性能的 Vector API。在推动整个生态系统发展方面，JavaOne 功不可没。

在软件工程师的职业生涯中，知识的“半衰期”通常为三年，这意味着我们掌握的一半知识在三年后将变得毫无价值。但这个行业本身就意味着不断充电与持续学习，技术大会或许是了解行业现状的有效途径。本书的诞生即源于 NFJS 巡回研讨会对作者的启发。NFJS 始于 2001 年，主要关注软件开发领域出现的最新技术，Java 与 JVM 是其中重点讨论的话题。

不过对 Java 开发者而言，掌握这门语言的各种技巧至关重要，夯实基础始终是首要任务。本书沿袭了 O'Reilly Media “编程食谱书”的一贯风格，将提炼自实际开发的问题以范例的形式展现给读者，使开发者对 Java 的关键知识点了然于心。O'Reilly Media 以出版技术类图书著称，其“动物书”系列与 Manning Publications 的“服饰书”系列备受开发人员的推崇。

非常感谢人民邮电出版社图灵文化发展有限公司的朱巍老师给予我翻译本书的机会，以及李冰、岳新欣、傅志红等各位编辑为本书所做的辛勤努力。虽然译者尽力而为，但水平有限，疏漏之处在所难免。恳请读者不吝赐教，提出宝贵的意见和建议。译者的联系方式：[milesjiang314@gmail.com](mailto:milesjiang314@gmail.com)。

蒋楠

2018年6月

# 序

毫无疑问，Java 8 引入的新特性（特别是 lambda 表达式和 Stream API）让这门语言经历了一次巨大的飞跃。多年来，我一直是 Java 8 的忠实用户，并在各种会议、研讨会以及博客上不遗余力地向开发人员介绍这些新特性。我很清楚，尽管 lambda 表达式和流让 Java 具备了更多函数式编程的特点（并行处理的威力也得以发挥），不过它们并非吸引开发人员的真正原因。新的习惯用法能让解决特定问题变得更为简单和高效，这才是 Java 8 备受追捧的根源所在。

作为一名程序员、演讲者和撰稿人，我不仅希望其他开发人员注意到 Java 语言的演变，还希望能展示这些演变是如何提高工作效率的。我们可以采用更简单的方法解决问题，甚至还能解决不同类型的问题。我之所以欣赏本书作者 Ken Kousen 的工作，是因为他在写作时严格遵循以下原则：帮助读者获取新知识，避免将时间花在已经了解或不需要的细节上。Ken 专注于对一线开发人员有价值的那些技术。

我第一次接触到 Ken 的工作，是他在 JavaOne 2013 会议上发表题为“Making Java Groovy: Simplify Your Java Development with Groovy”的演讲时。那时，我所在的团队正在为编写易读且有用的测试而殚精竭虑，我们所考虑的一种解决方案正是 Groovy。作为一名长期使用 Java 的程序员，我不愿意为了编写测试而去学习一门全新的语言，特别是我自认为已经了解如何编写测试时。然而，聆听 Ken 为 Java 程序员所做的 Groovy 介绍让我受益匪浅，他并未重复那些我已烂熟于心的内容，而是直入正题，使我迅速掌握了许多所需的知识。我意识到，选择合适的学习材料能极大地提高学习效率，我不必为了一个环节的应用而将一门语言的细枝末节全部吃透。因此，我立即购买了 Ken 撰写的 *Making Java Groovy* 一书<sup>1</sup>。

本书延续了类似的题材。作为经验丰富的开发者，我们无须像初学者一样学习 Java 8 和 Java 9 引入的所有新特性，也没有时间这样做。我们需要的是一本能迅速查找相关特性介绍的指南，并给出可以用于实际开发的示例。本书就是这样一本指南。书中范例来自开发人员在日常工作中遇到的问题，并介绍了如何利用 Java 8 和 Java 9 的新特性解决这些问

---

注 1：该书由 Manning Publications 于 2013 年 9 月出版，Ken 在 JavaOne 2013 会议上的演讲即以此为题。

——译者注

题，从而以更自然的方式让开发人员对这门语言的变化了然于心。我们可以举一反三，将所学的知识运用到实际开发中。

即便是 Java 8 和 Java 9 的长期使用者，依然可以从本书中受到启发。有关归约运算符的讨论切实加深了我对这种函数式编程风格的理解，而且我也无须重新理清思路。专门探讨 Java 9 新特性的章节正是开发人员所需要的，这些新特性尚未广为人知。本书提供了一种很好的方法，能够帮助读者快速有效地了解 Java 的最新发展。对所有希望提高自身知识水平的 Java 开发人员而言，本书堪称良师益友。

Trisha Gee

Java Champion

JetBrains 公司 Java 布道师

---

# 前言

## 与时俱进的Java

有时候，很难相信一门已保持了20年向后兼容性的语言会发生如此巨大的变化。在Oracle于2014年3月发布Java SE 8之前，作为最权威的服务器端编程语言，Java已然赢得“21世纪的COBOL”这一美誉。Java稳定且应用广泛，同时还不遗余力地追求性能。变化来得很慢，但还是来了。正因为如此，每当Java发布新版本时，企业的升级意愿并不迫切。

不过，在Java SE 8发布之后，一切都发生了改变。Java SE 8将“Lambda项目”（Project Lambda）纳入其中，这个重大的创新将函数式编程（functional programming）的概念引入这门杰出的面向对象语言。lambda表达式、方法引用以及流从根本上改变了Java的习惯用法。自此之后，开发人员一直在努力跟上这门语言前进的步伐。

本书无意评判这些变化能否对Java开发有所促进，也无意探讨是否可以通过其他途径实现同样的目的。本书只是告诉读者，新特性已经存在，我们应该如何利用它们完成工作。这也是本书采用范例形式编写的原因。读者可以根据需要阅读本书，了解Java引入的新特性将如何帮助自己实现既定目标。

换言之，一旦掌握这种新的程序设计模型，就能享受它所带来的诸多优点。函数式代码往往更简单，而且更易于编写和理解。函数式编程强调不可变性（immutability），这使得编写的并发代码更简洁，调试和运行更容易成功。在Java初登舞台时，摩尔定律仍然有效：处理器的速度大约每18个月就提高一倍。而如今性能提升的根本在于，即使是手机也已大部分配备了多个处理器。

由于Java非常注重保持向后兼容性，不少企业和开发人员在迁移到Java SE 8时并未采用新的习惯用法。即便如此，Java SE 8仍然是一个值得尝试的强大平台，而且Oracle已于2015年4月正式宣布停止对Java 7提供支持。

Java SE 8发布至今已有几年时间，大部分Java开发人员目前都已转向JDK 8。现在，深入了解Java SE 8对未来开发的意义和影响正当其时。希望本书能让这一过程变得更加容易。

# 目标读者

本书范例假定读者对 Java SE 8 之前的版本已有所了解。尽管不要求读者精通 Java，书中也会讨论某些较早的概念，但本书并非一本针对初学者的 Java 或面向对象编程教程。如果读者已使用 Java 开发过项目，并且熟悉标准库，阅读本书时应该不会感到困难。

本书涵盖与 Java SE 8 有关的几乎所有内容，并专门有一章介绍 Java 9 的新特性。如果希望了解 Java SE 8 新增的函数式习惯用法将如何改变代码的编写方式，这本包含丰富用例的教程是一个不错的选择。

Java 广泛应用于服务器端开发，拥有丰富的开源库和工具支持系统。Spring 和 Hibernate 是两种最流行的开源框架，二者只支持（或很快将只支持）Java 8 及以上的版本。如果读者计划使用 Java 8 或 Java 9 进行开发，本书讨论的范例或许能有所启发。

# 本书结构

本书以范例的形式编写和组织内容。但在讨论涉及 lambda 表达式、方法引用以及流的范例时，有时也会涉及其他内容。因此，前 6 章将介绍相关概念，不过读者无须以任何特定的顺序阅读。

各章主要内容如下。

- 第 1 章将介绍 lambda 表达式和方法引用的基础知识，然后讨论接口的新特性，包括默认方法和静态方法。此外，还将定义“函数式接口”，并解释它对于理解 lambda 表达式的重要性。
- 第 2 章主要介绍 Java 8 引入的 `java.util.function` 包，它包括 `Consumer`、`Supplier`、`Predicate` 以及 `Function` 这四类特殊的函数式接口，它们的应用贯穿于整个标准库。
- 第 3 章将介绍流的概念及其表示抽象的方法。流支持对数据进行转换和过滤，而非迭代地进行处理。这一章的范例将讨论与流相关的映射、过滤、归约等概念，它们与第 9 章介绍的并行和并发有密切的关系。
- 第 4 章主要介绍流数据的排序，并讨论如何将其转换为集合。这一章还将介绍分区和分组，它们将一般意义上的数据库操作转换为简单的库调用。
- 第 5 章是综合性的一章。在掌握 lambda 表达式、方法引用以及流的用法之后，读者将学习如何综合运用它们来解决某些有趣的问题。这一章还将讨论惰性、延迟执行、闭包复合等概念，以及异常处理这个令人头疼的问题。
- 第 6 章将讨论 Java 8 引入的颇具争议性的 `Optional` 类。这一章的范例将介绍 `Optional` 类的用法，以及如何创建实例并从中提取值。此外，我们将进一步讨论 `Optional` 类中 `map` 与 `flatMap` 操作所体现的函数式概念，以及它们与流中的 `map` 与 `flatMap` 操作有何不同。
- 第 7 章将介绍输入 / 输出流（与函数式流相对）的实际应用，以及 Java 8 针对文件和目录处理为标准库引入的一些函数式概念。
- 第 8 章将讨论 Java 8 引入的 Date-Time API，以及它如何取代传统且饱受诟病的 `Date` 类和 `Calendar` 类。这种新的 API 基于 `Joda-Time` 库，凝聚了大量开发人员多年的使用经验，已被重写为 `java.time` 包。坦率地讲，即便 Date-Time API 是 Java 8 新增的唯一特性，升级到 Java 8 也物有所值。

- 第 9 章主要介绍流模型的一种隐式承诺：通过一次方法调用，可以将串行流转换为并行流，从而充分利用计算机中所有可用的处理器。并发涉及的内容很多，这一章将重点介绍 Java 库的新增功能，这些功能便于用户进行试验，并评估成本和收益是否值得付出努力。
- 第 10 章将介绍 Java 9 引入的众多新特性，该版本于 2017 年 9 月 21 日正式发布。Jigsaw 本身的内容已可单独成书，但其基础概念十分清晰，这一章将对此进行介绍。其他范例将讨论接口中的私有方法，并介绍 Stream、Collectors 与 Optional 新增的各种方法，以及如何创建日期流<sup>1</sup>。
- 附录 A 将介绍 Java 中的泛型。泛型是 Java 1.5 引入的概念，但大部分开发人员对泛型只是略知皮毛，仅停留在完成工作所需的层面上。不过浏览 Java 8 和 Java 9 的 Javadoc 就会知道，这种日子已一去不复返。附录 A 旨在介绍如何阅读并解释 API，以帮助读者理解较为复杂的方法签名。

读者不必以任何特定的顺序阅读各章及其范例。各章之间互为补充，而且每个范例最后都包括指向其他范例的参考信息，所以从任何地方开始阅读均无不妥。章节分组是为了将相近的范例归类，但读者完全可以根据需要阅读所需的范例，以解决当前遇到的任何问题。

## 排版约定

本书使用以下排版约定。

- **黑体**  
表示新术语和重点强调的内容。
- 等宽字体 (*constant width*)  
表示程序片段，以及正文中出现的变量、函数名、数据库、数据类型、环境变量、语句和关键词等。
- 加粗等宽字体 (***constant width bold***)  
表示命令以及其他需要用户输入的文字。
- 等宽斜体 (*constant width italic*)  
表示这些值应该替换为用户输入，或根据上下文确定。



该图标表示提示或建议。

---

注 1：是的，我也希望能将讨论 Java 9 的章节排在第 9 章，不过仅仅为了偶然的对称性而对章节进行重新编排似乎不太合适。有这个脚注也就够了。



该图标表示一般性说明。



该图标表示警告。

## 示例代码

读者可以从 <http://www.it-ebooks.com.cn/book/2032> 下载本书源代码，它们分别存储 Java 8 的相关范例（第 10 章除外）、Java 9 的相关范例以及范例 9.7 讨论的复杂示例，且均已配置为包含测试与构建文件的 Gradle 项目。

本书旨在帮助读者解决开发中遇到的问题。一般而言，读者不必获得 O'Reilly 的授权，就可以在自己的程序或文档中使用书中的示例代码。不过如果需要大量复制代码，则应该联系我们以获得许可。例如，读者可以直接在程序中使用本书的代码块。但是，销售或分发 O'Reilly 图书的配套光盘则需要获得许可。引用本书及其示例代码来解答问题，不需要获得许可。如果在自己的产品文档中大量使用书中的示例代码，则需要获得许可。

欢迎读者在使用本书的示例代码时注明出处，但这不是强制要求。通常要注明书名、作者、出版社和 ISBN。例如，“*Modern Java Recipes* by Ken Kousen (O'Reilly). Copyright 2017 Ken Kousen, 978-0-491-97317-2”。

如果读者认为对示例代码的使用不在合理使用和上述无须授权的范围之内，那么请通过 [permissions@oreilly.com](mailto:permissions@oreilly.com) 联系我们。

## O'Reilly Safari



Safari（之前称为 Safari Books Online）是一个为企业、政府、教育机构以及个人提供培训的会员制在线学习平台。

Safari 会员可以访问 250 多家出版商提供的数千种图书、培训视频、学习路径、互动教程以及精选列表等资源，这些出版商包括 O'Reilly Media、Harvard Business Review、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Adobe、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett、Course Technology 等。

# 联系我们

如果读者对本书有任何评论或疑问，请通过以下地址联系我们。

美国：

O'Reilly Media, Inc.  
1005 Gravenstein Highway North  
Sebastopol, CA 95472

中国：

奥莱利技术咨询（北京）有限公司  
北京市西城区西直门南大街2号成铭大厦C座807室（100035）

有关本书的技术性问题和建议，请通过邮箱 [bookquestions@oreilly.com](mailto:bookquestions@oreilly.com) 联系我们。

欢迎访问 O'Reilly Media 网站，获取更多的图书、课程、会议信息以及最新动态。

我们的 Facebook 地址是 <http://facebook.com/oreilly>。

请关注我们的 Twitter 动态：<http://twitter.com/oreillymedia>。

我们的 YouTube 视频地址是 <http://www.youtube.com/oreillymedia>。

## 致谢

2015年7月下旬，我与NFJS巡回研讨会<sup>2</sup>负责人 Jay Zimmerman 进行了一次谈话，这次谈话意外促成了本书的诞生。我当时是（目前也仍然是）NFJS巡回研讨会的成员，那一年 Venkat Subramaniam 博士<sup>3</sup>做了多次关于 Java 8 的演讲。由于 Subramaniam 博士决定在未来一年减少演讲的次数，Jay 询问我是否有兴趣从2016年初开始发表类似的演讲。我从20世纪90年代中期起就一直采用 Java 编写程序（所使用的第一个版本为 Java 1.0.6），对研究新 API 也抱有浓厚兴趣。念及此，我答应了 Jay 的请求。

几年来，我针对 Java 引入的函数式特性做了多次介绍。2016年秋天，在完成上一本范例教程<sup>4</sup>的写作后，我决定继续为 O'Reilly Media 撰写一本讨论 Java 的范例教程。那时我想当然地认为，这项工作应该是信手拈来。

---

注 2：NFJS 巡回研讨会（No Fluff Just Stuff Software Symposium Tour）于 2001 年在美国丹佛创办，重点关注现代软件开发与架构领域出现的最新技术和最佳实践，演讲者包括作者、咨询师、开发人员以及行业专家。从 2001 年至今，NFJS 巡回研讨会已经在全美各地举办了 500 多场活动，参与人数超过 8 万人。——译者注

注 3：Subramaniam 博士是 Agile Developer 公司创始人，休斯敦大学兼职教授。作为敏捷开发领域的权威人士，他培训了世界各地数以千计的软件开发人员。他还是一位多产的技术图书作者，所撰写的《Groovy 程序设计》一书是 Java 程序员学习 Groovy 的不二之选。另著有《高效程序员的 45 个习惯》《JavaScript 测试驱动开发》等书。——译者注

注 4：Gradle Recipes for Android，由 O'Reilly Media 出版，该书讨论了 Gradle 构建工具在 Android 项目中的应用。