O'REILLY®

TURING 图灵程序设计丛书

Java 8 函数式编程

Java 8 Lambdas: Functional Programming for the Masses

[英] Richard Warburton 著
王群锋 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 8函数式编程

Java 8 Lambdas
Functional Programming For The Masses

[英] Richard Warburton 著
王群锋 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • 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（CIP）数据

Java 8函数式编程 / (英) 沃伯顿 (Warburton, R.) 著 ; 王群锋译. -- 北京 : 人民邮电出版社, 2015.4
（图灵程序设计丛书）
ISBN 978-7-115-38488-1

Ⅰ. ①J… Ⅱ. ①沃… ②王… Ⅲ. ①JAVA语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2015)第025390号

内 容 提 要

多年以来，函数式编程被认为是少数人的游戏，不适合推广给普罗大众。写作此书的目的就是为了挑战这种思想。本书将探讨如何编写出简单、干净、易读的代码；如何简单地使用并行计算提高性能；如何准确地为问题建模，并且开发出更好的领域特定语言；如何写出不易出错，并且更简单的并发代码；如何测试和调试 Lambda 表达式。

如果你已经掌握 Java SE，想尽快了解 Java 8 新特性，写出简单干净的代码，那么本书不容错过。

◆ 著 [英] Richard Warburton
译 王群锋
责任编辑 李松峰
执行编辑 李 静 仇祝平
责任印制 杨林杰

◆ 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
北京鑫正大印刷有限公司印刷

◆ 开本：800×1000 1/16
印张：9.25
字数：191千字 2015年4月第1版
印数：1－3 500册 2015年4月北京第1次印刷
著作权合同登记号 图字：01-2014-6949号

定价：39.00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：京崇工商广字第 0021 号

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

版权声明

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。无论是通过书籍出版，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：‘如果你在路上遇到岔路口，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*

前言

多年以来，函数式编程被认为是少数人的游戏，这些人总是强调自己在智力上的优越性，认为函数式编程的智慧不适合推广给普罗大众。写作此书的目的就是为了挑战这种思想，函数式编程并没有多么了不起，也绝不是少数人的游戏。

在过去的两年中，我请伦敦 Java 社区的开发人员以各种方式测试 Java 8 的新特性。我发现很多人都喜欢 Java 8 的新用法和类库。他们有可能被一些术语和高大上的概念吓到，但是稍稍一丁点儿函数式编程技巧都能给编程带来便利，他们对此喜不自胜。人们津津乐道的话题之一是使用新的 Stream API 操作对象和集合类时（比如从所有的唱片列表中过滤出在英国本地出品的唱片时），代码是多么易读。

组织这些 Java 社区活动，让我认识到了示例代码的重要性。人们通过不断地阅读和消化这些简单的示例，最终归纳出某种模式。我还意识到术语是多么令人讨厌，因此，在介绍一个晦涩的概念时，我都会给出通俗易懂的解释。

对很多人来说，Java 8 提供的函数式编程元素有限：没有单子[1]，没有语言层面的惰性求值，也没有为不可变性提供额外支持。对实用至上的程序员来说，这没什么大不了的，我们只想在类库级别抽象，写出简单干净的代码来解决业务问题。如果有人为我们写出这样的类库，那再好不过了，这样我们就可以把主要精力放在日常工作上了。

为什么要阅读本书

本书将探讨如下主题。

- 如何编写出简单、干净、易读的代码——尤其是对于集合的操作？
- 如何简单地使用并行计算提高性能？

注 1：别担心，这是本书唯一提及单子的地方。

- 如何准确地为问题建模，并且开发出更好的领域特定语言？
- 如何写出不易出错，并且更简单的并发代码？
- 如何测试和调试 Lambda 表达式？

将 Lambda 表达式加入 Java，并不只是为了提高开发人员的生产效率，业界也对这一特性有根本性的需求。

本书读者对象

本书面向那些已经掌握 Java SE，并且想尽快了解 Java 8 新特性的开发人员。

如果你对 Lambda 表达式感兴趣，想知道它怎么帮助你提升专业技能，那么这本书就是为你而写的。我假设读者还不知道 Lambda 表达式，以及 Java 8 中核心类库的变化，我将从零开始介绍这些概念、类库和技术。

虽然我想让所有开发人员都来买这本书，但这不现实，这不是一本适合所有人的书。如果你一点儿也不懂 Java，那么这本书就不适合你。同时，尽管本书会详细讲解 Java 中的 Lambda 表达式，但是我不会解释怎样在其他语言中使用 Lambda 表达式。

我也不会讲解 Java SE 中一些基本的概念，比如集合类、匿名内部类或者 Swing 中的事件处理机制。我假设读者已经掌握了这些知识。

怎样阅读本书

本书采用了示例驱动的写作风格：介绍完一个概念之后，就会紧跟一段代码。代码中的一些片段，有时你可能无法全部看懂。没关系，通常在代码后面会紧跟一段文字，讲解代码的细节。

这种方式能让你边学边练，多数章节还在最后提供了练习题，供读者自行练习。我强烈建议读者读完一章后完成这些练习，熟能生巧。每个务实的程序员都知道，自欺欺人很容易，你觉得读懂一段代码了，其实还是遗漏了一些细节。

使用 Lambda 表达式，就是将复杂性抽象到类库的过程。在本书中，我会引入很多常用类库的细节。第 2 章至第 6 章介绍了 JDK 8 中核心语言的变化以及升级后的类库。

最后三章介绍了如何在真实环境下使用函数式编程。第 7 章介绍一些让测试和调试 Lambda 表达式变得容易的技巧；第 8 章解释现有的那些良好的软件设计原则如何应用到 Lambda 表达式上；第 9 章讨论并发，怎样使用 Lambda 表达式写出易读且易于维护的并发代码。涉及第三方类库时，这些章节也会加以介绍。

读者可以将前四章当作 Java 8 的入门指南——要用好 Java 8，每个人都必须学会这些知识。

后面的几章难度略高，但掌握了这几章的内容，你就可以成为知识更加全面的程序员，在自己的设计中得心应手地使用 Lambda 表达式。你在不断学习的过程中，也会接触大量的练习，答案可以在 GitHub（https://github.com/RichardWarburton/java-8-Lambdas-exercises）上找到。如果你能边学边练，就能迅速掌握 Lambda 表达式。

本书排版规范

本书使用以下排版规范。

- 楷体
  表示新术语。
- 等宽字体
  表示程序片段，也用于在正文中表示程序中使用的变量、函数名、数据库、数据类型、环境变量、语句和关键字等元素。
- 等宽粗体
  表示应该由用户逐字输入的命令或者其他文本。
- 等宽斜体
  表示将由用户提供的值（或由上下文确定的值）替换的文本。

这个图标表示提示或建议。

这个图标表示重要说明。

这个图标表示警告或提醒。

使用代码示例

可以在这里下载本书随附的资料（代码示例、练习题等）：https://github.com/RichardWarburton/java-8-lambdas-exercises。

让本书助你一臂之力。也许你需要在自己的程序或文档中用到本书中的代码。除非大段大段地使用，否则不必与我们联系取得授权。例如，无需请求许可，就可以用本书中的几段代码写成一个程序。但是销售或者发布 O'Reilly 图书中代码的光盘则必须事先获得授权。引用书中的代码来回答问题也无需授权。将大段的示例代码整合到你自己的产品文档中则必须经过许可。

使用我们的代码时，希望你能标明它的出处，但不强求。出处信息一般包括书名、作者、出版商和书号，例如：*Java 8 Lambdas*，Richard Warburton 著（O'Reilly，2014）。版权所有，978-1-449-37077-0。

如果还有关于使用代码的未尽事宜，可以随时与我们联系：permissions@oreilly.com。

Safari® Books Online

Safari Books Online（http://www.safaribooksonline.com）是应需而变的数字图书馆。它同时以图书和视频的形式出版世界顶级技术和商务作家的专业作品。

Safari Books Online 是技术专家、软件开发人员、Web 设计师、商务人士和创意人士开展调研、解决问题、学习和认证培训的第一手资料。

对于组织团体、政府机构和个人，Safari Books Online 提供各种产品组合和灵活的定价策略。用户可通过一个功能完备的数据库检索系统访问 O'Reilly Media、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、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 以及其他几十家出版社的上千种图书、培训视频和正式出版之前的书稿。要了解 Safari Books Online 的更多信息，我们网上见。

联系我们

请把对本书的评价和问题发给出版社。

美国：

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

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室（100035）

奥莱利技术咨询（北京）有限公司

O'Reilly 的每一本书都有专属网页，你可以在那儿找到本书的相关信息，包括勘误表、示例代码以及其他信息。本书的网站地址是：

http://oreil.ly/java_8_lambdas。

对于本书的评论和技术性问题，请发送电子邮件到：

bookquestions@oreilly.com

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：

http://www.oreilly.com

我们在 Facebook 的地址如下：http://facebook.com/oreilly

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

我们的 YouTube 视频地址如下：http://www.youtube.com/oreillymedia

致谢

虽然本书的封面上署的是我的名字，但本书得以出版要归功于很多人。

首先要感谢我的编辑 Meghan 和 O'Reilly 的出版团队，他们让整个出版过程变得很愉快，而且他们还适当加快了本书的出版进度。还要感谢 Martijn 和 Ben 将我引荐给 Meghan，没有这次会面就不会有这本书。

审阅过程极大地提升了本书的质量，衷心感谢那些正式或非正式参与审阅的朋友，他们是：Martijn Verburg、Jim Gough、John Oliver、Edward Wong、Brian Goetz、Daniel Bryant、Fred Rosenberger、Jaikiran Pai 和 Mani Sarkar。尤其要感谢 Martijn，他给了我如何写一本技术书的实战指导。

如果忘记感谢 Oracle 公司的 Project Lambda 项目组，我不会原谅自己。更新一个成熟的语言是一项巨大的挑战，他们不辱使命，我也因此有了得以编写本书的素材。在 Java 8 发布早期版本时，伦敦的 Java 社区积极参与测试，通过这些测试，很容易就发现了开发人员犯了哪类错误，哪些地方可以修复，感谢他们！

在写作本书的过程中，我得到了很多人的支持和帮助，特别是我的父母。在我需要的时候，他们总是陪伴在身边。我的朋友们也总是给我积极的评价和鼓励，包括 Compsoc 里的那些老伙计们，特别是 Sadiq Jaffer 和基督少年军，感谢你们！

目录

第1章

简介

在开始探索 Lambda 表达式之前，首先我们要知道它因何而生。本章将介绍 Lambda 表达式产生的原因，以及本书的写作动机和组织结构。

1.1 为什么需要再次修改Java

1996 年 1 月，Java 1.0 发布，此后计算机编程领域发生了翻天覆地的变化。商业发展需要更复杂的应用，大多数程序都跑在功能强大的多核 CPU 的机器上。带有高效运行时编译器的 Java 虚拟机（JVM）的出现，使程序员将更多精力放在编写干净、易于维护的代码上，而不是思考如何将每一个 CPU 时钟周期、每字节内存物尽其用。

多核 CPU 的兴起成为了不容回避的事实。涉及锁的编程算法不但容易出错，而且耗费时间。人们开发了 `java.util.concurrent` 包和很多第三方类库，试图将并发抽象化，帮助程序员写出在多核 CPU 上运行良好的程序。很可惜，到目前为止，我们的成果还远远不够。

开发类库的程序员使用 Java 时，发现抽象级别还不够。处理大型数据集合就是个很好的例子，面对大型数据集合，Java 还欠缺高效的并行操作。开发者能够使用 Java 8 编写复杂的集合处理算法，只需要简单修改一个方法，就能让代码在多核 CPU 上高效运行。为了编写这类处理批量数据的并行类库，需要在语言层面上修改现有的 Java：增加 Lambda 表达式。

当然，这样做是有代价的，程序员必须学习如何编写和阅读使用 Lambda 表达式的代码，但是，这不是一桩赔本的买卖。与手写一大段复杂、线程安全的代码相比，学习一点新语法和一些新习惯容易很多。开发企业级应用时，好的类库和框架极大地降低了开发时间和成本，也为开发易用且高效的类库扫清了障碍。

对于习惯了面向对象编程的开发者来说，抽象的概念并不陌生。面向对象编程是对数据进行抽象，而函数式编程是对行为进行抽象。现实世界中，数据和行为并存，程序也是如此，因此这两种编程方式我们都得学。

这种新的抽象方式还有其他好处。不是所有人都在编写性能优先的代码，对于这些人来说，函数式编程带来的好处尤为明显。程序员能编写出更容易阅读的代码——这种代码更多地表达了业务逻辑的意图，而不是它的实现机制。易读的代码也易于维护、更可靠、更不容易出错。

在写回调函数和事件处理程序时，程序员不必再纠缠于匿名内部类的冗繁和可读性，函数式编程让事件处理系统变得更加简单。能将函数方便地传递也让编写惰性代码变得容易，惰性代码在真正需要时才初始化变量的值。

Java 8 还让集合类可以拥有一些额外的方法：`default` 方法。程序员在维护自己的类库时，可以使用这些方法。

总而言之，Java 已经不是祖辈们当年使用的 Java 了，嗯，这不是件坏事。

1.2 什么是函数式编程

每个人对*函数式编程*的理解不尽相同。但其核心是：在思考问题时，使用不可变值和函数，函数对一个值进行处理，映射成另一个值。

不同的语言社区往往对各自语言中的特性孤芳自赏。现在谈 Java 程序员如何定义函数式编程还为时尚早，但是，这根本不重要！我们关心的是如何写出好代码，而不是符合函数式编程风格的代码。

本书将重点放在函数式编程的实用性上，包括可以被大多数程序员理解和使用的技术，帮助他们写出易读、易维护的代码。

1.3 示例

本书中的示例全部都围绕一个常见的问题领域构造：音乐。具体来说，这些示例代表了在专辑上常常看到的信息，有关术语定义如下。

- `Artist`
  创作音乐的个人或团队。
  - `name`：艺术家的名字（例如“甲壳虫乐队”）。
  - `members`：乐队成员（例如“约翰·列侬”），该字段可为空。
  - `origin`：乐队来自哪里（例如“利物浦”）。