Jolt获奖图书的最新版本 78条程序员必知的黄金法则

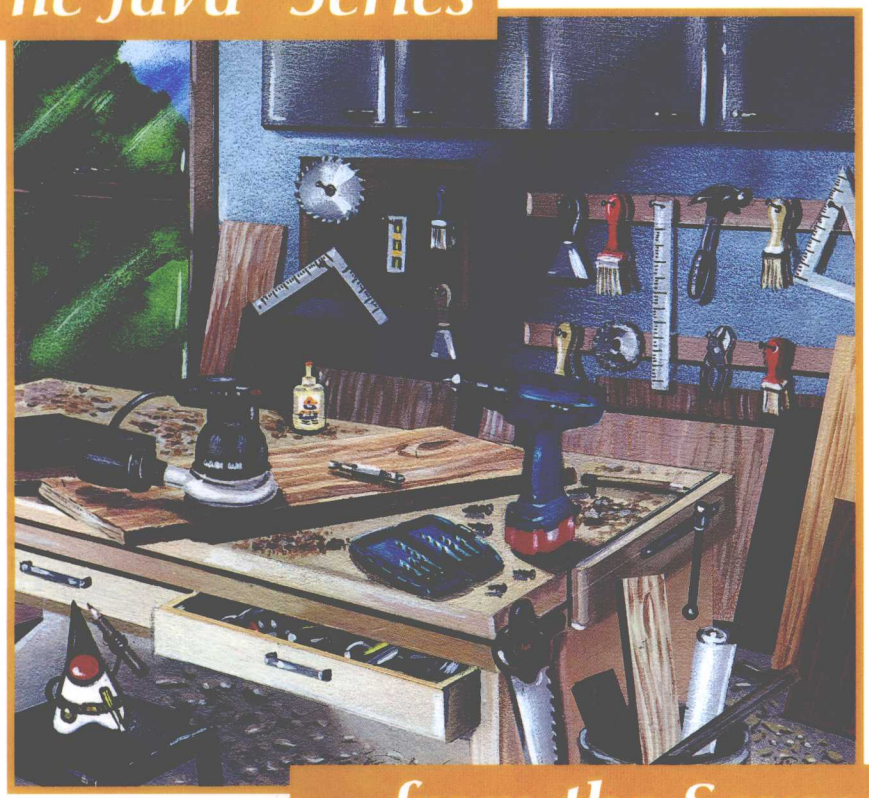
助您编写出更优秀代码的经典读本

Effective Java

（第2版）（英文版）

[美] Joshua Bloch 著

The Java™ Series



...from the Source

Sun microsystems

We're the dot in .com™

人民邮电出版社
POSTS & TELECOM PRESS

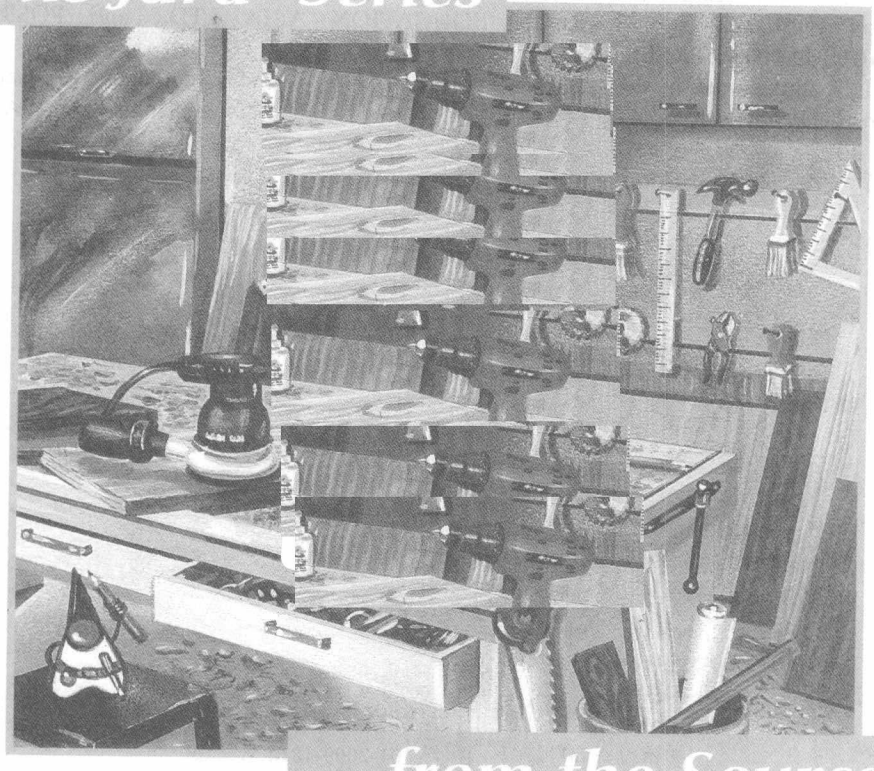
Effective Java

Effective Java

（第2版）（英文版）

[美] Joshua Bloch 著

The Java™ Series



...from the Source

人民邮电出版社
北京

图书在版编目（CIP）数据

Effective Java英文版：第2版：英文/（美）布洛克（Bloch, J.）著. —北京：人民邮电出版社，2009.9
ISBN 978-7-115-21131-6

Ⅰ. E… Ⅱ. 布… Ⅲ. JAVA语言—程序设计—英文
Ⅳ. TP312

中国版本图书馆CIP数据核字（2009）第124259号

Effective Java（第2版）（英文版）

◆ 著 [美] Joshua Bloch
责任编辑 李 际
执行编辑 赵 轩

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京顺义振华印刷厂印刷

◆ 开本：800×1000 1/16
印张：22.5
字数：498千字 2009年9月第1版
印数：1－2 000册 2009年9月北京第1次印刷
著作权合同登记号 图字：01-2009-1814号

ISBN 978-7-115-21131-6

定价：39.00元

读者服务热线：(010)67132705 印装质量热线：(010)67129223
反盗版热线：(010)67171154

内容提要

本书介绍了在 Java 编程中 78 条经典的、实用性极高的经验规则，这些经验规则可以帮助开发人员来解决每天都有可能面对的大多数问题。书中对 Java 平台设计专家所使用的技术的介绍，揭示了如何事半功倍地的编写清晰、健壮和高效的代码。本书是经典图书 Effective Java 的第 2 版，涵盖了 Java 5 中的重要变化，并删除了一些过时的内容。本书所介绍的每条规则，都以简明易懂的语言来阐释，并通过示例代码进一步加以说明。

本书内容翔实，层次分明，是一本可以帮助技术人员更深层次理解 Java 的参考用书。

Praise for the First Edition

"I sure wish I had this book ten years ago. Some might think that I don't need any Java books, but I need this one."

—James Gosling, fellow and vice president, Sun Microsystems, Inc., and inventor of the Java programming language

"An excellent book, crammed with good advice on using the Java programming language and object-oriented programming in general."

—Gilad Bracha, distinguished engineer, Cadence Design Systems, and coauthor of *The Java™ Language Specification, Third Edition* (Addison-Wesley, 2005)

"10/10—anyone aspiring to write good Java code that others will appreciate reading and maintaining should be required to own a copy of this book. This is one of those rare books where the information won't become obsolete with subsequent releases of the JDK library."

—Peter Tran, bartender, JavaRanch.com

"The best Java book yet written.... Really great; very readable and eminently useful. I can't say enough good things about this book. At JavaOne 2001, James Gosling said, 'Go buy this book!' I'm glad I did, and I couldn't agree more."

—Keith Edwards, senior member of research staff, Computer Science Lab at the Palo Alto Research Center (PARC), and author of *Core JINI* (Prentice Hall, 2000)

"This is a truly excellent book done by the guy who designed several of the better recent Java platform APIs (including the Collections API)."

—James Clark, technical lead of the XML Working Group during the creation of the XML 1.0 Recommendation; editor of the XPath and XSLT Recommendations

"Great content. Analogous to Scott Meyers's classic *Effective C++*. If you know the basics of Java, this has to be your next book."

—Gary K. Evans, OO mentor and consultant, Evanetics, Inc.

"In my estimation, no more than one good programming book appears per year (and there are certainly many years that saw none): Knuth's trilogy, the K&R White Book, Kernighan and Plauger's *Software Tools*... Bloch's book fits in well with this august company. Get it."

—Andrew Binstock, *Software Development Times*, August 15, 2001

"This is a superb book. It clearly covers many of the language/platform subtleties and trickery you need to learn to become a real Java master."

—Victor Wiewiorowski, vice president development and code quality manager, ValueCommerce Co., Tokyo, Japan

"I like books that under-promise in their titles and over-deliver in their contents. This book has 57 items of programming advice that are well chosen. Each item reveals a clear, deep grasp of the language. Each one illustrates in simple, practical terms the limits of programming on intuition alone, or taking the most direct path to a solution without fully understanding what the language offers."

—Michael Ernest, Inkling Research, Inc.

"Every bit of this book is essential for Java designers. Reading this book before you start delivering products can easily repay its cost thousands of times."

—Richard Mateosian, IEEE micro, July/August 2002 (vol. 22, no. 4)

"Great how-to resource for the experienced developer."

—John Zukowski, author of numerous Java books

"I picked this book up two weeks ago and can safely say I learned more about the Java language in three days of reading than I did in three months of study! An excellent book and a welcome addition to my Java library."

—Jane Griscti, I/T advisory specialist

To my family: Cindy, Tim, and Matt

Acknowledgments

Acknowledgments for the Second Edition

I thank the readers of the first edition of this book for giving it such a kind and enthusiastic reception, for taking its ideas to heart, and for letting me know what a positive influence it had on them and their work. I thank the many professors who used the book in their courses, and the many engineering teams that adopted it.

I thank the whole team at Addison-Wesley for the their kindness, professionalism, patience, and grace under pressure. Through it all, my editor Greg Doench remained unflappable: a fine editor and a perfect gentleman. My production manager, Julie Nahil, was everything that a production manager should be: diligent, prompt, organized, and friendly. My copy editor, Barbara Wood, was meticulous and tasteful.

I have once again been blessed with the best team of reviewers imaginable, and I give my sincerest thanks to each of them. The core team, who reviewed every chapter, consisted of Lexi Baugher, Cindy Bloch, Beth Bottos, Joe Bowbeer, Brian Goetz, Tim Halloran, Brian Kernighan, Rob Konigsberg, Tim Peierls, Bill Pugh, Yoshiki Shibata, Peter Stout, Peter Weinberger, and Frank Yellin. Other reviewers included Pablo Bellver, Dan Bloch, Dan Bornstein, Kevin Bourrillion, Martin Buchholz, Joe Darcy, Neal Gafter, Laurence Gonsalves, Aaron Greenhouse, Barry Hayes, Peter Jones, Angelika Langer, Doug Lea, Bob Lee, Jeremy Manson, Tom May, Mike McCloskey, Andriy Tereshchenko, and Paul Tyma. Again, these reviewers made numerous suggestions that led to great improvements in this book and saved me from many embarrassments. And again, any remaining embarrassments are my responsibility.

I give special thanks to Doug Lea and Tim Peierls, who served as sounding boards for many of the ideas in this book. Doug and Tim were unfailingly generous with their time and knowledge.

I thank my manager at Google, Prabha Krishna, for her continued support and encouragement.

Finally, I thank my wife, Cindy Bloch, for encouraging me to write, for reading each item in raw form, for helping me with Framemaker, for writing the index, and for putting up with me while I wrote.

Foreword

IF a colleague were to say to you, "Spouse of me this night today manufactures the unusual meal in a home. You will join?" three things would likely cross your mind: third, that you had been invited to dinner; second, that English was not your colleague's first language; and first, a good deal of puzzlement.

If you have ever studied a second language yourself and then tried to use it outside the classroom, you know that there are three things you must master: how the language is structured (grammar), how to name things you want to talk about (vocabulary), and the customary and effective ways to say everyday things (usage). Too often only the first two are covered in the classroom, and you find native speakers constantly suppressing their laughter as you try to make yourself understood.

It is much the same with a programming language. You need to understand the core language: is it algorithmic, functional, object-oriented? You need to know the vocabulary: what data structures, operations, and facilities are provided by the standard libraries? And you need to be familiar with the customary and effective ways to structure your code. Books about programming languages often cover only the first two, or discuss usage only spottily. Maybe that's because the first two are in some ways easier to write about. Grammar and vocabulary are properties of the language alone, but usage is characteristic of a community that uses it.

The Java programming language, for example, is object-oriented with single inheritance and supports an imperative (statement-oriented) coding style within each method. The libraries address graphic display support, networking, distributed computing, and security. But how is the language best put to use in practice?

There is another point. Programs, unlike spoken sentences and unlike most books and magazines, are likely to be changed over time. It's typically not enough to produce code that operates effectively and is readily understood by other persons; one must also organize the code so that it is easy to modify. There may be ten ways to write code for some task T. Of those ten ways, seven will be awkward, inefficient, or puzzling. Of the other three, which is most likely to be similar to the code needed for the task T' in next year's software release?

There are numerous books from which you can learn the grammar of the Java Programming Language, including *The Java™ Programming Language* by Arnold, Gosling, and Holmes [Arnold05] or *The Java™ Language Specification* by Gosling, Joy, yours truly, and Bracha [JLS]. Likewise, there are dozens of books on the libraries and APIs associated with the Java programming language.

This book addresses your third need: customary and effective usage. Joshua Bloch has spent years extending, implementing, and using the Java programming language at Sun Microsystems; he has also read a lot of other people's code, including mine. Here he offers good advice, systematically organized, on how to structure your code so that it works well, so that other people can understand it, so that future modifications and improvements are less likely to cause headaches—perhaps, even, so that your programs will be pleasant, elegant, and graceful.

Guy L. Steele Jr.
Burlington, Massachusetts
April 2001

前　言

在我于2001年写了本书的第1版之后，Java平台又发生了很多变化，我想是该出第2版的时候了。Java 5中新增加了泛型、枚举类型、注解、自动装箱和for-each循环。此外还增加了新的并发类库：java.util.concurrent。我有幸和Gilad Bracha一起，带领团队设计了最新的语言特性，并参加了设计和开发并发类库的团队，这个团队是由Dong Lea领导的。

Java平台中另一个大的变化是广泛采用了现代的IDE（Integrated Development Environment），例如，Eclipse、IntelliJ IDEA和NetNeans，以及静态分析工具的IDE，如FindBugs。虽然我还未参与到这部分工作，但已经从中受益匪浅，并且很清楚它们对于Java开发体验所带来的影响。

2004年，我到了Google公司工作，但在过去的4年中，我仍然继续参与了Java平台的开发，在Google公司和JCP（Java Community Process）的大力帮助下，继续并发和集合API的开发。我还有幸利用Java平台去开发供Google内部使用的类库，从而了解了作为一名用户的感受。

我在2001年编写本书第1版时，主要目的是与读者分享我的经验，让大家避免我所走过的弯路，使大家更容易成功。本版仍然大量采用来自Java平台类库的真实范例。

第1版所带来的反应远远超出了我的预期。我在收集所有新的资料以使本书保持最新时，尽可能地保持了资料的真实。毫无疑问，本书的篇幅从以前的57个条目发展到了78条目。我不仅增加了23个条目，并且修改了原来的所有资料，并删去了一些已经过时的条目。

我在第1版的前言中说过：Java程序设计语言和它的类库非常有益于代码质量和效率的提高，并且使得Java进行编码成为一种乐趣。最新版本中的变化使得Java平台日趋完善。现在这个平台比2001年的要大得多，也复杂得多，但是一旦掌握了使用新特性的模式和用法，它们就会使你的程序变得更完美，使你的工作变得更轻松。我希望本版能够体现出我对Java平台持续的热情，并将这种热情传递给你，帮助你更加高效和愉快地使用Java平台及其新的特性。

Joshua Bloch
San Jose，California
2008年4月

Contents

CHAPTER 1

Introduction

THIS book is designed to help you make the most effective use of the Java™ programming language and its fundamental libraries, `java.lang`, `java.util`, and, to a lesser extent, `java.util.concurrent` and `java.io`. The book discusses other libraries from time to time, but it does not cover graphical user interface programming, enterprise APIs, or mobile devices.

This book consists of seventy-eight items, each of which conveys one rule. The rules capture practices generally held to be beneficial by the best and most experienced programmers. The items are loosely grouped into ten chapters, each concerning one broad aspect of software design. The book is not intended to be read from cover to cover: each item stands on its own, more or less. The items are heavily cross-referenced so you can easily plot your own course through the book.

Many new features were added to the platform in Java 5 (release 1.5). Most of the items in this book use these features in some way. The following table shows you where to go for primary coverage of these features:

Feature	Chapter or Item
Generics	Chapter 5
Enums	Items 30–34
Annotations	Items 35–37
For-each loop	Item 46
Autoboxing	Items 40, 49
Varargs	Item 42
Static import	Item 19
`java.util.concurrent`	Items 68, 69