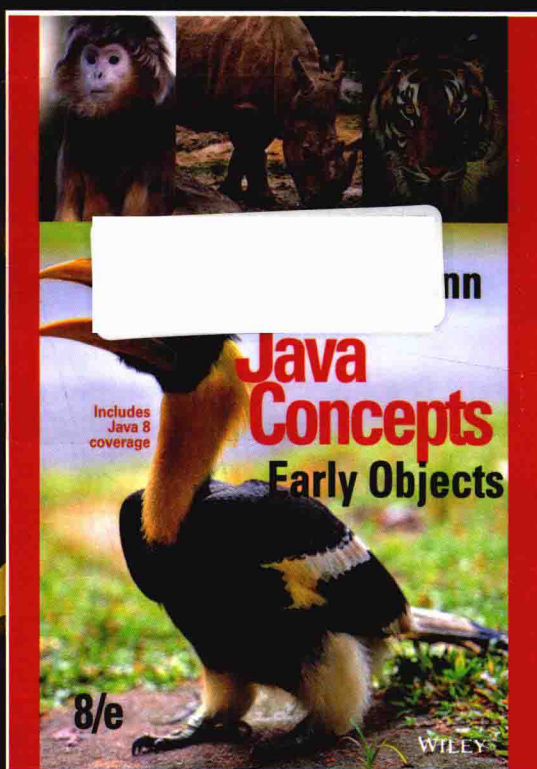


Java程序设计概念 对象先行

[美] 凯·霍斯特曼 (Cay Horstmann) 著
林琪 肖斌 等译

Java Concepts: Early Objects
Eighth Edition



计 算 机 科 学 丛 书

原书第8版

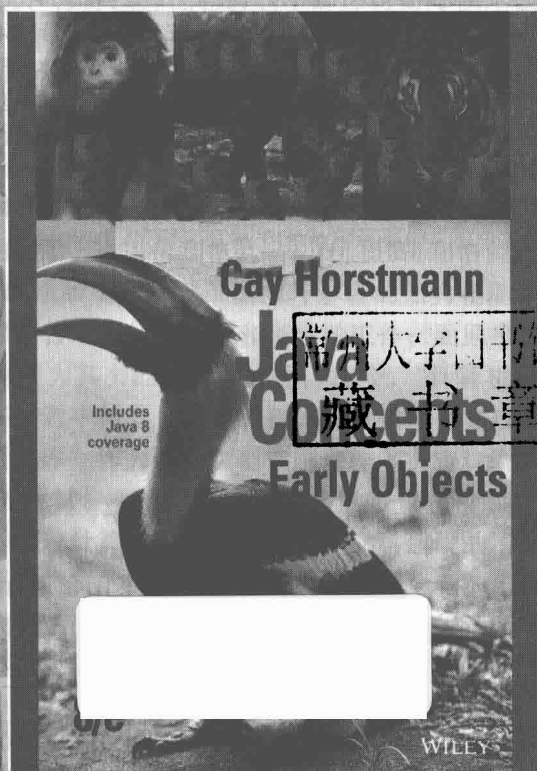
Java程序设计概念

对象先行

[美] 凯·霍斯特曼 (Cay Horstmann) 著

林琪 肖斌 等译

Java Concepts: Early Objects
Eighth Edition



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

Java 程序设计概念: 对象先行 (原书第 8 版) / (美) 凯·霍斯特曼 (Cay Horstmann) 著; 林琪等译. —北京: 机械工业出版社, 2018.3

(计算机科学丛书)

书名原文: Java Concepts: Early Objects, Eighth Edition

ISBN 978-7-111-59366-9

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

中国版本图书馆 CIP 数据核字 (2018) 第 046926 号

本书版权登记号: 图字 01-2017-3406

Copyright © 2015 John Wiley & Sons.

All rights reserved. This translation published under license. Authorized translation from the English language edition, entitled Java Concepts: Early Objects, Eighth Edition, ISBN 9781119056454, by Cay Horstmann, Published by John Wiley & Sons, New York, Chichester, Brisbane, Singapore and Toronto. No part of this book may be reproduced in any form without the written permission of John Wiley & Sons Inc.

本书中文简体字版由约翰·威利父子公司授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

本书封底贴有 Wiley 防伪标签, 无标签者不得销售。

这是一本关于 Java 和计算机编程的入门教材, 书中重点关注基础知识, 并强调如何有效地学习。本书共分三部分。第一部分 (第 1 ~ 7 章) 是基础知识, 首先简要介绍了计算机科学和 Java 编程, 展示了如何管理预定义的类以及如何根据给定的规范定义类, 之后讲解了基本数据类型、分支、循环和数组; 第二部分 (第 8 ~ 12 章) 讲解面向对象设计, 包括类的设计、继承和多态、接口、异常处理和基本文件输入/输出, 以及两个重要的案例研究; 第三部分 (第 13 ~ 15 章) 是在线内容, 讲解了数据结构与算法, 具体包括递归、排序和查找、链表、集合、映射、栈和队列。

本书非常适合作为计算机科学家、工程师和其他学科学生的第一本编程教材。

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: 关 敏

责任校对: 李秋荣

印 刷: 北京瑞德印刷有限公司

版 次: 2018 年 5 月第 1 版第 1 次印刷

开 本: 185mm×260mm 1/16

印 张: 40.5

书 号: ISBN 978-7-111-59366-9

定 价: 119.00 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

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzsj@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，不仅擘划了研究的范畴，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

这是一本让人惊喜的 Java 入门书。

它不仅能让你轻松进入 Java 世界，更重要的是，通过学习本书，能够把面向对象的设计理念自然而然地融入你的学习和编程中。“对象先行”正是本书倡导的一个重要方面，这也是本书区别于其他教材的一个突出特点。我们很多人最早学习 Java 时都经历了从面向过程到面向对象的概念转换，由于前期养成了一些不好的编程习惯，所以往往饱受纠正错误的痛苦。如果那时能有这样一本书，我们会少走很多弯路。从这个意义上讲，你真的很幸运！

本书的目的是介绍基本概念，为 Java 编程奠定基础。本书分为基础知识、面向对象设计以及数据结构和算法三部分，看似平常，但内容绝不枯燥。每一章每一节都精心设计，以通俗易懂的方式清晰地讲解概念，更通过实用示例强化理解。另外，作为第一线的教师，本书作者深谙教学之道，将课堂设计中的多种教学元素很好地贯穿在了本书中。每一章首先给出学习目标，并提供大量提示和详细说明来强调重点，这些内容就像我们上课时所记的笔记，能帮助我们真正掌握所学的内容。不仅如此，本书的版式也让人眼前一亮，不同层次的读者可以按照学习要求灵活地安排学习内容和顺序，更能感受学习的乐趣。

本书特别设置了“计算与社会”“专题”及“科学和商业练习”等板块，来提高学生的学习兴趣。更重要的是，通过这些贴近生活的实用主题，书中介绍的知识不再只是生硬的书本概念，而直接对应于切切实实的具体实践，这会让学习过程更加有的放矢。还有一点需要指出的是，本书中的练习覆盖面很广，可能涉及一些对你来说很生疏的领域，以至于让人望而却步，但这些练习并不要求你掌握那些领域的理论背景，它们只是用来提供一些“场景”，使你能够在这些场景下实际操练你学到的 Java 编程知识。

希望本书能为你开启 Java 编程世界的大门，帮你踏出坚实的第一步。

本书由林琪和肖斌主译，朱涛江、杜志秀、杨春生、刘亮、王宇、张伶、王志淋、刘跃邦参与了全书的修改整理，程芳、吴忠望、张练达、陈峰、杨健康、姚勇、张莹、牛亚峰、高强、赵蓓、荆涛、刘恒完善了关键部分的翻译。全体人员共同完成了本书的翻译工作。不过由于水平有限，译文肯定有不当之处，敬请批评指正。

林琪

2017 年 11 月



这是一本关于 Java 和计算机编程的入门教材，书中重点关注基础知识，并强调如何有效地学习。本书面向有不同兴趣和能力层次不同的广大学生，非常适合作为计算机科学家、工程师和其他学科学生的第一本编程教材。本书不要求读者有任何编程经验，只需要掌握一定的高中代数知识就可以学习。

本书有以下主要特点。

对象先行，渐进地讲授面向对象知识。

第 2 章将讲解如何使用标准库中的对象和类。第 3 章会介绍根据给定规范实现类的基本机制，这样学生就可以在掌握分支、循环和数组的同时使用简单的对象。面向对象设计将从第 8 章开始介绍。这种渐进的方法可以让学生在研究核心算法有关内容的整个过程中使用对象，而不会养成一些坏习惯，否则以后还得花很大力气才能改掉这些坏习惯。

指导和实用示例可以帮助学生达成学习目标。

初学的程序员通常会问“怎么开始？现在我要做什么？”当然，作为一种十分复杂的活动，编程肯定无法简化成一组菜谱式的指令。不过，逐步指导对于建立自信很有帮助，还有助于为当前任务提供一个大致轮廓。这种“操作指南”可以帮助学生完成常见的编程任务。我们还在线提供了另外一些实用示例。

明确指出解决问题的策略。

实用、循序渐进的技术讲解可以帮助学生为编程问题设计解决方案并做出评价。书中会在合适的位置介绍解决问题的策略，这些策略可以帮助学生克服障碍，获得成功。具体包括：

- 算法设计（利用伪代码）
- 跟踪对象
- 先手动处理（手写演算）
- 流程图
- 选择测试用例
- 手动跟踪
- 故事板
- 先解决比较简单的问题
- 调整算法
- 通过管理实体对象发现算法
- 对象数据模式
- 递归地思考
- 估计一个算法的运行时间

实践锻造完美。

当然，学习编程的学生还要能够实现不太简单的程序，不过首先他们要有能够成功的信心。本书每一节最后都有大量自查问题。“实践”环节指出了每一节之后可以尝试的练习。另外我们还在线提供了额外的实践机会，包括一些自动评分的编程练习和强调技能的多选题。

形象直观的方法促进并方便读者阅读。

照片以类比的方式形象地解释了计算机概念的内涵和行为。循序渐进的图表详细展示了复杂的程序操作。语法框和示例表以一种紧凑的格式提供了大量典型和特殊的用例。借助这些视觉手段，在集中精力学习文字内容之前就能对大致情况有所了解。

强调基础，同时保证技术准确。

对于刚开始学习编程的人来说，百科全书式的面面俱到并没有太大帮助，不过也不能矫枉过正，将内容缩减为一个过于简化的要点列表对初学者同样没有好处。本书会成块地介绍基础知识，方便读者消化吸收。另外，在读者希望获得更多信息时，我们会提供单独的说明，更深入地介绍一些好的实践做法或语言特性。这里不会人为地过于简化来给人以知识的幻觉。

加强工程实践。

书中关于软件质量和常见错误给出了大量很有用的提示，这些将鼓励读者养成良好的编程习惯。可选的测试活动强调测试驱动开发，鼓励学生系统地测试程序。

提供一个可选的图形活动。

图形是对象的一个非常好的例子。很多学生喜欢编写创建图形或使用图形用户界面的程序。如果需要，可以使用第 2 章、第 3 章和第 10 章最后材料在课程中加入这些内容。

包含可选的科学和商业练习。

每一章最后都会针对科学和商业领域的问题给出一些高级练习。设计这些练习是为了让学生更多地参与，这些练习体现了编程在应用领域的作用和意义。

本版新增内容

针对 Java 8 全面更新

Java 8 引入了很多令人兴奋的新特性，本书这一版已经全面更新来利用这些特性。现在接口可以有默认和静态方法，另外由于引入了 `lambda` 表达式，所以可以很容易地为只有一个方法的接口提供实例。关于接口的一章和介绍排序的章节也已经更新，以便根据需要利用这些新特性。

另外，如 `try-with-resources` 语句等 Java 7 特性会放在正文中介绍。

交互式学习

Codecheck（代码检查）是一个创新的在线服务，学生可以利用这个服务解决编程问题。你可以布置已准备的练习，也可以很容易地增加你自己的练习。有关的更多内容请访问 <http://codecheck.it>。

本书结构

本书很自然地分为三大部分，如图 1 所示。各章的组织与前一版同样灵活；图中还展示了各章之间的依赖关系。

第一部分 基础知识（第 1 章～第 7 章）

第 1 章是对计算机科学和 Java 编程的一个简要介绍。第 2 章展示如何管理预定义类的



视觉手段帮助读者轻松阅读

© Terraxplorer/iStockphoto.

对象。第3章中介绍根据给定的规范构建你自己的简单类。第4章~第7章介绍基本数据类型、分支、循环和数组。

第二部分 面向对象设计 (第8章~第12章)

第8章用系统的方式开始介绍类设计这一主题,还会介绍UML记法的一个非常简单的子集。关于多态和继承的讨论分为两章。第9章介绍继承和多态,第10章介绍接口。异常处理和基本文件输入/输出在第11章介绍。异常层次体系提供了继承的一个很有用的例子。第12章提供了面向对象设计的一个介绍,还包括两个重要的案例研究。

第三部分 数据结构与算法 (第13章~第15章)

第13章~第15章(本书网站上提供)^①是对算法和数据结构的介绍,具体包括递归、排序和查找、链表、集合、映射、栈和队列。一学期的课程可能无法全面介绍这些主题。不过,可以在第7章之后根据需要选学(见图1)。第13章介绍递归,首先从简单的例子开始,逐步过渡到更有意思的应用(这些应用采用迭代方式实现可能很困难)。第14章介绍二分排序算法和归并排序,还会非正式地介绍大O记法。各个数据结构会在标准Java集合库上下文中介绍。你会了解标准库的基本抽象(如迭代器、集合和映射),还会了解不同集合的性能特点。

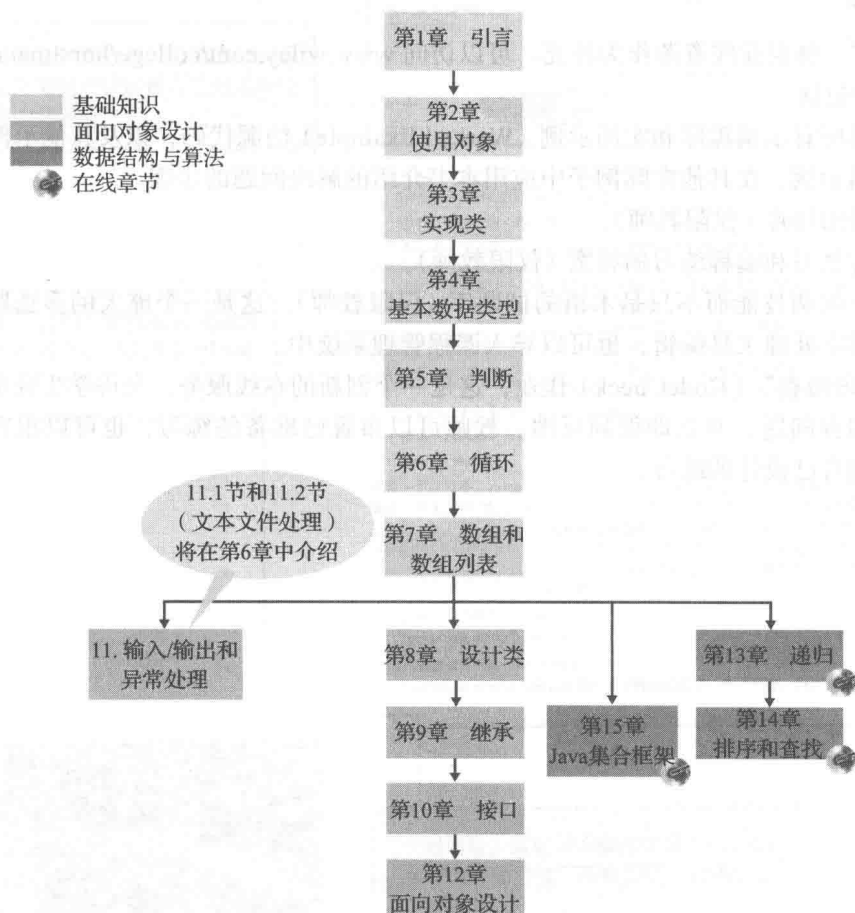


图1 各章之间的依赖关系

① 见华章公司网站：www.hzbook.com。——编辑注

附录

很多教师发现要求所有作业采用一致的风格会有很多好处。不过，如果附录 E 中的风格指南与教师的偏好或当地的习惯有出入，可以使用这个风格指南的电子版并相应地修改。附录 F ~ 附录 J 可以在网上找到。

A. Unicode 基本拉丁语和拉丁语 -1 子集

B. Java 操作符小结

C. Java 保留字小结

D. Java 库

E. Java 语言编码指南

F. 工具小结

G. 数字系统

H. UML 小结

I. Java 语法小结

J. HTML 小结

Web 资源

本书有一整套在线资源作为补充。可以访问 www.wiley.com/college/horstmann 在线配套网站，其中包括：

- 本书所有示例程序和实用示例 (Worked Example) 的源代码，以及其他示例程序。
- 实用示例，在其他实际例子中应用本书介绍的解决问题的步骤。
- 课程幻灯片 (仅限教师)。
- 所有复习和编程练习的答案 (仅限教师)。
- 一个强调技能而不只是术语的试题库 (仅限教师)。这是一个庞大的多选题题集，可以用字处理工具编辑，也可以导入课程管理系统中。
- “代码检查” (CodeCheck) 作业，这是一个创新的在线服务，允许学生完成其中给出的编程问题，并立即得到反馈。教师可以布置已准备的练习，也可以很容易地增加他们自己设计的练习。

本书提供了多种教学元素，这些元素共同作用，强调并重视编程的关键概念和基本原则，另外还提供了一些提示和详细说明，通过合理地组织，这些部分可以支持和深化基础知识。除了传统要素（如各章学习目标和大量的练习）之外，每一章还包含一些视觉元素，这是特意重视视觉的学习者设计的。

12 第一部分 基础知识

加注解的语法框会对新语言构造提供一个简要、形象的概要描述。

注解会解释必要的组成部分，并指出常见错误或这种语法有关最佳实践的更多信息。

每一章中，会用加灰底的形式指出这里将引入新概念，并对主要思想提供一个概要说明。

语法 1.1 Java 程序

每个Java程序都包含一个main方法，这个方法首部如下。

```
public class HelloPrinter
{
    public static void main(String[] args)
    {
        System.out.println("Hello, World!");
    }
}
```

程序运行时执行main方法中的语句。

每个程序至少包含一个类。选择一个能描述程序动作的类名。

编写你自己的程序时替换这个语句。

每个语句以一个分号结束。见常见错误1.1。

确保开始和结束大括号匹配。

在我们的示例程序中，main方法只有一个语句：

```
System.out.println("Hello, World!");
```

这个语句会打印一行文本，即“Hello, World!”。在这个语句中，我们调用（call）了一个方法，这个方法由一个相当长的名字 System.out.println 指定，使用这个方法名的原因在这里先不做解释。

我们不用实现这个方法，写Java库的程序员已经为我们写好了这个方法实现。我们只是希望这个方法能完成它预期的任务，也就是打印一个值。

在Java中调用一个方法时，需要指定：

1. 你想要使用的方法（在这里就是 System.out.println）。

2. 这个方法完成其任务所需的所有值（在这里就是“Hello, World!”）。这样一个值的相应术语是参数（argument）。参数用小括号包围。多个参数之间用逗号分隔。

用引号包围的字符串序列

“Hello, World!”

称为一个字符串（string）。必须将字符串的内容包围在引号中，这样编译器才能知道你确实要表示字面上的“Hello, World!”。之所以有这个要求，是因为假设你想打印单词 main，通过把它包围在引号里（“main”），编译器知道你是指字符串 main，而不是名为 main 的方法。一般规则是必须把所有文本字符串用引号包围，这样编译器会把它们看作纯文本，而不会试图将它们解释为程序指令。

也可以打印数值。例如，以下语句

```
System.out.println(3 + 4);
```

会计算表达式 3 + 4，并显示数字 7。

System.out.println 方法会打印一个字符串或数字，然后换行。例如，下面的语句序列：



与计算机程序中的变量类似，停车位也有标识和内容。

利用与日常对象的类比解释概念的性质和行为，如变量、数据类型、循环等。

让人记忆深刻的照片会强化类比，帮助学生记住这些概念。



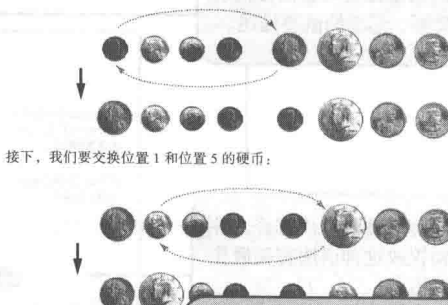
(左) © jchamp/iStockphoto; (中) © StevenCarrieJohnson/iStockphoto; (右) © jsmith/iStockphoto.

解决问题部分讲授提出想法和评价所提出方案的一些技术，通常会使用纸笔或其他工具。这些部分会强调计划和解决问题的主要工作，使学生可以成功地应用计算机解决问题。

你可以摆出一些硬币，来尝试这3种操作，感受一下。

这对于我们的问题（交换数组的前一半和后一半）有什么帮助？

下面把第一个硬币放在正确的位置上，让它与第5个硬币交换。不过，作为Java程序员，我们会说我们要交换位置0和位置4的硬币：



接下，我们要交换位置1和位置5的硬币：

操作指南 6.1

写一个循环



这个操作指南会帮助你了解实现一个循环语句的过程。我们将利用以下示例问题介绍相应的步骤。

问题陈述 读取12个温度（分别对应12个月），并显示最高温度的月份。例如，根据worldclimate.com，死谷（Death Valley）的平均最高温度如下（按月的顺序，单位为℃）：

18.2 22.6 26.4 31.1 36.6 42.2 45.7 44.5 40.2 33.1 24.2 17.6

在这种情况下，有最高温度（45.7℃）的月份是7月，所以程序要显示7。

步骤 1

确定在循环内部要完成的工作

每个循环需要做一些重复性的工作，如：

- 读取另一项。
- 更新一个值（如一个银行余额或总额）。
- 递增一个计数器。

如果不确定需要在循环里做什么，先写出手解决这个问题时你可能采取的步

实用示例 6.1

信用卡处理



了解如何使用循环去除信用卡号中的空格。



操作指南为常见的编程任务提供一个循序渐进的指南，特别强调计划和测试。它们会回答初学者的问题，“现在我要做什么”，另外在解决问题的步骤序列中会结合关键概念。

实用示例将操作指南中的步骤应用到一个不同的例子，展示如何使用这些步骤计划、实现和测试另一个编程问题的解决方案。

表 2-1 Java 中的变量声明

变 量 名	说 明
<code>int width = 20;</code>	声明一个整型变量，并初始化为 20
<code>int perimeter = 4 * width;</code>	初始值不一定是个固定值（当然，width 必须已经声明）
<code>String greeting = "Hi!";</code>	这个变量的类型为 String，初始化为字符串 "Hi!"
<code>height = 30;</code>	错误：缺少类型。这个语句不是一个声明，而是将一个新值赋给一个已有的变量，见 2.2.5 节
<code>int width = "20";</code>	错误：不能将一个数字变量初始化为字符串 "20"（注意引导）
<code>int width;</code>	声明一个整型变量，但是没有初始化这个变量。这可能会带来错误。参见常见错误 2.1
<code>int width, height;</code>	在一个语句中声明两个整型变量。本书中，会用单独的语句声明各个变量

表为初学者提供多个具体的示例。这些表会指出常见的错误，同时对该节的内容提供一个快速参考。

赋值操作符 = 不表示数学上的相等。

这表示“计算 $height + 10$ 的值，并把这个值存储在变量 $width$ 中”。

在 Java 编程语言中， $=$ 操作符指示一个动作，就是改变一个变量的值。这种用法与数学中使用等号 ($=$) 表示相等的用法是不同的。例如，在 Java 中，以下语句是完全合法的：

```
width = width + 10;
```

这表示“计算 $width + 10$ 的值①，并把这个值存储在变量 $width$ 中②”（参见图 2-4）。

在 Java 中，变量 $width$ 用在等号两边没有任何问题。当然，在数学中，方程 $width = width + 10$ 是无解的。

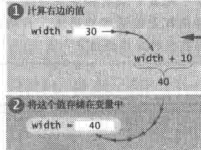


图 2-4 执行语句 $width = width + 10$

进度图会跟踪代码段的执行，帮助学生查看程序流。

这个 for 循环很清楚地将初始化、条件和更新表达式组织在一起。但是，有一点很重要，要认识到这些表达式并不是一起执行的（参见图 6-3）。

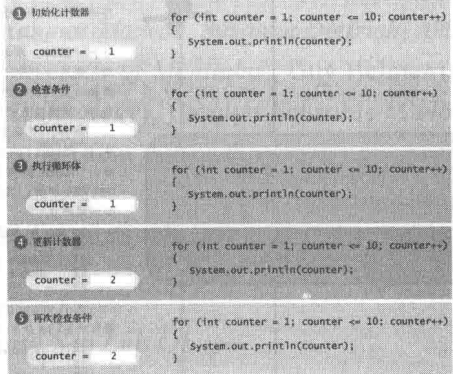


图 6-3 for 循环的执行

- 初始化只在进入循环之前执行一次。①
- 条件在每次迭代前检查。② ⑤
- 更新在每次迭代后执行。④

每一节最后的自查练习有助于学生仔细思考新学的内容——另外可以在课堂讲授中让学生展开讨论。

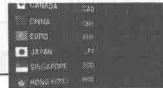


11. 将 `Investment` 类的 for 循环写为一个 while 循环。
12. 这个循环打印多少个？

```
for (int n = 10; n >= 0; n--) {
    System.out.println(n);
}
```
13. 编写一个 for 循环，打印 10 ~ 20（包括 10 和 20）之间的所有偶数。
14. 编写一个 for 循环，计算 1 ~ n 的所有整数之和。
15. 如何修改 `InvestmentRunner.java` 程序打印 20, 40, ..., 100 年后的余额？

实践：现在你可以试着完成本章最后的以下练习：R6.7, R6.13, E6.9, E6.13。

- 商业 E6.20 货币转换。编写一个程序，首先让用户输入今天 1 美元兑换日元的汇率，然后读入美元值，将各个美元值转换为日元。使用 0 作为分隔。



- 科学 P6.17 放射性物质的放射性衰变可以用以下公式建模：

$$A = A_0 e^{-(\ln 2) t / h}$$

式中， A 为时间 t 的物质数量， A_0 为时间 0 的物质数量， h 为半衰期。
 Te-99 是一种放射性同位素，在大脑透视中使用。它的半衰期为 6 小时。你的程序应当显示患者在注射显影剂之后 24 小时内每小时患者体内的相对量 A/A_0 。



可选的科学和商业练习使学生能够完成 Java 的实际应用。

section_1/Investment.java

```
1 /**
2  * A class to monitor the growth of an investment that
3  * accumulates interest at a fixed annual rate.
4  */
5  public class Investment
6  {
7      private double balance;
8      private double rate;
9      private int year;
10
11     /**
12      * Constructs an Investment object from a starting balance and
13      * interest rate.
14      * @param aBalance the starting balance
15      * @param aRate the interest rate in percent
16      */
17     public Investment(double aBalance, double aRate)
18     {
19         balance = aBalance;
20         rate = aRate;
21         year = 0;
22     }
23
24     /**
25      * Keeps accumulating interest until a target balance has
26      * been reached.
27      * @param targetBalance the desired balance
28      */
```

程序清单经过仔细设计以方便读者阅读。

常见错误描述学生常犯的错误，并解释为什么会出现这些错误，以及如何处理。

常见错误 7.4



长度和大小

遗憾的是，Java 中确定数组、数组列表和字符串中元素个数的语法不一致。一个常见的错误就是混淆这些语法。必须记住每种数据类型的正确语法。

数据类型	元素个数
数组	a.length
数组列表	a.size()
字符串	a.length()

编程技巧 5.5



手动跟踪

要了解一个程序是否正确地工作，有一个非常有用的技术，称为手动跟踪。你在一张纸上模拟程序的活动。可以对伪代码或 Java 代码使用这种方法。

拿出一张索引卡、一张餐巾纸或者你能拿到的任何一张纸。为每个变量建立一列。准备好程序代码。使用一个标记（如一个回形针）标记当前语句。在你心里一次执行一个语句。每次一个变量的值改变时，划掉原来的值，并在原来的值下面写上新值。

例如，下面利用前面运行程序得到的数据来跟踪 getTax 方法。构造 TaxReturn 对象时，income 实例变量设置为 80 000，status 设置为 MARRIED。然后调用 getTax 方法。在 TaxReturn.java 的第 31 行和第 32 行中，tax1 和 tax2 初始化为 0。

```
29 public double getTax()  
30 {  
31     double tax1 = 0;  
32     double tax2 = 0;  
33 }
```

由于 status 不是 SINGLE，所以转向外部 if 语句的 else 分支（第 46 行）。

```
34 if (status == SINGLE)  
35 {  
36     if (income <= RATE1_SINGLE_LIMIT)
```



手动跟踪帮助你了解一个程序是否正确地工作。

income	status	tax1	tax2
80000	MARRIED	0	0

编程技巧会解释好的编程实践，并鼓励学生利用手动跟踪等技巧和技术更高效地编程。

专题介绍一些可选的主题，并提供其他主题的解释。

专题 11.2



文件对话框

在一个有图形用户界面的程序中，需要程序用户选择一个文件时，你可能希望使用一个对话框（如下图所示的对话框）。JFileChooser 类为 Swing 用户界面工具包实现了一个文件对话框。

JFileChooser 类有很多选项，可以很精细地调整对话框的显示，不过最基本的形式相当简单：构造一个文件选择器对象；然后调用 showOpenDialog 或 showSaveDialog 方法。这两个方法会显示同样的对话框，不过选择文件的按钮标签分别为“Open”和“Save”，这取决于你调用哪一个方法。

要在屏幕上更好地放置对话框，可以指定弹出对话框的用户界面组件。如果你不关心对话框在哪里弹出，可以简单地传入 null。showOpenDialog 和 showSaveDialog 方法会返回 JFileChooser.APPROVE_OPTION（如果用户选择了一个文件）或 JFileChooser.CANCEL_OPTION（如果用户取消了选择）。

Java 8 说明 10.4

8

lambda 表达式

在 10.4 节中，你已经看到如何使用接口指定行为的变化。average 方法需要度量各个对象，这是通过调用所提供的 Measurer 对象的 measure 方法做到的。

遗憾的是，average 方法的调用者必须做大量的工作；具体包括：定义一个实现 Measurer 接口的类，还要构造这个类的一个对象。Java 8 为这些步骤提供了一个方便的快捷方法，前提是这个接口只有一个抽象方法。这种接口称为函数接口（functional interface），因为它的作用就是定义一个函数。Measurer 接口就是函数接口的一个例子。

要指定这个函数，可以使用一个 lambda 表达式，这个表达式采用一种紧凑的记法定义方法的参数和返回值。下面给出一个例子：

```
(Object obj) -> ((BankAccount) obj).getBalance()
```

Java 8 说明提供 Java 8 中新增特性的详细说明。



计算与社会 6.1 数字盗版

如果你读到这里，说明你已经写过一些计算机程序，已经亲身体会到即使编写最普通的程序也要花很多时间。要编写一个真正的软件产品，如一个财务应用或一个计算机游戏，要花费大量时间和资金。如果没有机会从中挣更多的钱，很少有人愿意投入这些时间和资金，更少有人愿意这么做。（确实，有些公司会免费发布软件，希望用户单击广告或者升级到更全面的付费版本。另外一些公司会免费发布允许用户阅读和使用文件的软件，但是创建这些文件的软件则要付费购买。最后，还有一些人会出于热情并投入时间开发允许你免费复制的

版权法和专利，以鼓励新产品的开发。容忍盗版泛滥的国家会发现，他们会得到大量廉价的外来软件，但没有本地制造商愿意为他们自己的人民设计好的软件，如使用本地文字的字处理器，或者适合本地税法的财务程序。

最初出现一个巨大的软件市场时，开发商因盗版导致的损失愤愤不平。他们希望反击，通过使用各种机制，如软件狗（软件运行之前必须在打印机端口上加装的一个设备），确保只有真正合法的所有者能够使用软件。但是，合法用户很讨厌这些措施。他们已经为软件付费，还必须

计算与社会介绍有关计算的社会和历史主题，一方面是满足学生的兴趣，另一方面也是为了符合 ACM/IEEE 课程大纲的“历史和社会背景”要求。

非常感谢 John Wiley & Sons 公司的 Bryan Gambrel、Don Fowley、Jenny Welter、Jessy Moor、Jennifer Lartz、Billy Ray 和 Tim Lindner，以及 Publishing Services 的 Vickie Piercey，感谢他们对这个项目的帮助。尤其要向 Cindy Johnson 致以诚挚的感谢，感谢她的辛勤工作、睿智判断以及对细节无与伦比的专注。

非常感谢 Jose Cordova (*The University of Louisiana at Monroe*, Suzanne Dietrich, *Arizona State University, West Campus*), Mike Domaratzki (*University of Manitoba*), Guy Helmer (*Iowa State University*), Peter Lutz (*Rochester Institute of Technology*), Carolyn Schauble (*Colorado State University*), Brent Seales (*University of Kentucky*), 和 Brent Wilson (*George Fox University*) 对补充材料的突出贡献。

非常感谢审校这一版手稿、提出宝贵建议以及找出大量错误和遗漏的所有人，他们包括：

Robin Carr, *Drexel University*

Gerald Cohen, *The Richard Stockton College of New Jersey*

Aaron Keen, *California Polytechnic State University, San Luis Obispo*

Aurelia Smith, *Columbus State University*

Aakash Taneja, *The Richard Stockton College of New Jersey*

Craig Tanis, *University of Tennessee at Chattanooga*

Katherine Winters, *University of Tennessee at Chattanooga*

每一个新版本都建立在之前版本审校人员和使用者的建议及体验基础之上。非常感谢所有人做出的非凡贡献：

Eric Aaron, *Wesleyan University*

James Agnew, *Anne Arundel
Community College*

Tim Andersen, *Boise State University*

Ivan Bajic, *San Diego State University*

Greg Ballinger, *Miami Dade College*

Ted Bangay, *Sheridan Institute
of Technology*

Ian Barland, *Radford University*

George Basham, *Franklin University*

Jon Beck, *Truman State University*

Sambit Bhattacharya, *Fayetteville
State University*

Rick Birney, *Arizona State University*

Paul Bladek, *Edmonds Community
College*

Matt Boutell, *Rose-Hulman Institute of
Technology*

Joseph Bowbeer, *Vizrea Corporation*

Timothy A. Budd, *Oregon State
University*

John Bundy, *DeVry University Chicago*

Robert P. Burton, *Brigham Young
University*

Frank Butt, *IBM*

Jerry Cain, *Stanford University*

Adam Cannon, *Columbia University*

Michael Carney, *Finger Lakes
Community College*

Christopher Cassa, *Massachusetts
Institute of Technology*

Nancy Chase, *Gonzaga University*

Dr. Suchindran S. Chatterjee, *Arizona
State University*

Archana Chidanandan, *Rose-Hulman
Institute of Technology*

Vincent Cicirello, *The Richard Stockton
College of New Jersey*

Teresa Cole, *Boise State University*

Deborah Coleman, *Rochester Institute
of Technology*

Tina Comston, *Franklin University*

- Lennie Cooper, *Miami Dade College*
 Jose Cordova, *University of Louisiana, Monroe*
 Valentino Crespi, *California State University, Los Angeles*
 Jim Cross, *Auburn University*
 Russell Deaton, *University of Arkansas*
 Geoffrey Decker, *Northern Illinois University*
 H. E. Dunsmore, *Purdue University*
 Robert Duvall, *Duke University*
 Sherif Elfayoumy, *University of North Florida*
 Eman El-Sheikh, *University of West Florida*
 Henry A. Etlinger, *Rochester Institute of Technology*
 John Fendrich, *Bradley University*
 David Freer, *Miami Dade College*
 John Fulton, *Franklin University*
 David Geary, *Sabreware, Inc.*
 Margaret Geroch, *Wheeling Jesuit University*
 Ahmad Ghafarian, *North Georgia College & State University*
 Rick Giles, *Acadia University*
 Stacey Grasso, *College of San Mateo*
 Jianchao Han, *California State University, Dominguez Hills*
 Lisa Hansen, *Western New England College*
 Elliotte Harold
 Eileen Head, *Binghamton University*
 Cecily Heiner, *University of Utah*
 Guy Helmer, *Iowa State University*
 Ed Holden, *Rochester Institute of Technology*
 Brian Howard, *Depauw University*
 Lubomir Ivanov, *Iona College*
 Norman Jacobson, *University of California, Irvine*
 Steven Janke, *Colorado College*
 Curt Jones, *Bloomsburg University*
 Mark Jones, *Lock Haven University of Pennsylvania*
 Dr. Mustafa Kamal, *University of Central Missouri*
 Mugdha Khaladkar, *New Jersey Institute of Technology*
 Gary J. Koehler, *University of Florida*
 Elliot Koffman, *Temple University*
 Ronald Krawitz, *DeVry University*
 Norm Krumpe, *Miami University Ohio*
 Jim Leone, *Rochester Institute of Technology*
 Kevin Lillis, *St. Ambrose University*
 Darren Lim, *Siena College*
 Hong Lin, *DeVry University*
 Kathy Liszka, *University of Akron*
 Hunter Lloyd, *Montana State University*
 Youmin Lu, *Bloomsburg University*
 Kuber Maharjan, *Purdue University College of Technology at Columbus*
 John S. Mallozzi, *Iona College*
 John Martin, *North Dakota State University*
 Jeanna Matthews, *Clarkson University*
 Patricia McDermott-Wells, *Florida International University*
 Scott McElfresh, *Carnegie Mellon University*
 Joan McGrory, *Christian Brothers University*
 Carolyn Miller, *North Carolina State University*
 Sandeep R. Mitra, *State University of New York, Brockport*
 Teng Moh, *San Jose State University*
 Bill Mongan, *Drexel University*
 John Moore, *The Citadel*
 Jose-Arturo Mora-Soto, *Jesica Rivero-Espinosa, and Julio-Angel Cano-Romero, University of Madrid*
 Faye Navabi, *Arizona State University*
 Parviz Partow-Navid, *California State University, Los Angeles*
 George Novacky, *University of Pittsburgh*
 Kevin O'Gorman, *California Polytechnic State University, San Luis Obispo*
 Michael Olan, *Richard Stockton College*
 Mimi Opkins, *California State University Long Beach*
 Derek Pao, *City University of Hong Kong*
 Kevin Parker, *Idaho State University*

Jim Perry, *Ulster County Community College*
 Cornel Pokorny, *California Polytechnic State University, San Luis Obispo*
 Roger Priebe, *University of Texas, Austin*
 C. Robert Putnam, *California State University, Northridge*
 Kai Qian, *Southern Polytechnic State University*
 Cyndi Rader, *Colorado School of Mines*
 Neil Rankin, *Worcester Polytechnic Institute*
 Brad Rippe, *Fullerton College*
 Pedro I. Rivera Vega, *University of Puerto Rico, Mayaguez*
 Daniel Rogers, *SUNY Brockport*
 Chaman Lal Sabharwal, *Missouri University of Science and Technology*
 Katherine Salch, *Illinois Central College*
 John Santore, *Bridgewater State College*
 Javad Shakib, *DeVry University*
 Carolyn Schauble, *Colorado State University*
 Brent Seales, *University of Kentucky*
 Christian Shin, *SUNY Geneseo*
 Charlie Shu, *Franklin University*
 Jeffrey Six, *University of Delaware*
 Don Slater, *Carnegie Mellon University*
 Ken Slonneger, *University of Iowa*
 Donald Smith, *Columbia College*
 Joslyn A. Smith, *Florida International*

University
 Stephanie Smullen, *University of Tennessee, Chattanooga*
 Robert Strader, *Stephen F. Austin State University*
 Monica Sweat, *Georgia Institute of Technology*
 Peter Stanchev, *Kettering University*
 Shannon Tauro, *University of California, Irvine*
 Ron Taylor, *Wright State University*
 Russell Tessier, *University of Massachusetts, Amherst*
 Jonathan L. Tolstedt, *North Dakota State University*
 David Vineyard, *Kettering University*
 Joseph Vybihal, *McGill University*
 Xiaoming Wei, *Iona College*
 Jonathan S. Weissman, *Finger Lakes Community College*
 Todd Whittaker, *Franklin University*
 Robert Willhoft, *Roberts Wesleyan College*
 Lea Wittie, *Bucknell University*
 David Womack, *University of Texas at San Antonio*
 David Woolbright, *Columbus State University*
 Tom Wulf, *University of Cincinnati*
 Catherine Wyman, *DeVry University*
 Arthur Yanushka, *Christian Brothers University*
 Qi Yu, *Rochester Institute of Technology*
 Salih Yurttas, *Texas A&M University*

特色内容

Java Concepts: Early Objects (Eighth Edition)

章号和章名



常见错误



操作指南

和

实用示例



1 引言	缺少分号 单词拼写错误	用伪代码描述算法 编写一个铺地板瓷砖的算法
2 使用对象	使用未声明或未初始化的变量 混淆变量声明和赋值语句 试图如调用方法一样调用构造函数	你活了多少天? 处理图片
3 实现类	将构造函数声明为 void 忽略参数变量 在局部变量中重复实例变量 提供不必要的实例变量 忘记在构造函数中初始化对象引用	实现一个类 建立一个简单的菜单 绘制图形
4 基本数据类型	意外的整除 小括号不匹配	完成计算 计算金字塔的体积和表面积 计算旅行时间
5 判断	if 条件后有分号 使用 == 比较字符串 空悬 else 问题 结合多个关系操作符 混淆 && 和 条件	实现一个 if 语句 抽取中间字符
6 循环	不要想“我们到了吗?” 无限循环 差一错误	写一个循环 信用卡处理 处理图像中的像素 调试 示例调试会话
7 数组和数组列表	边界错误 未初始化和未填充的数组 低估数据集的大小 长度和大小	处理数组 掷骰子 世界人口表

内容可以从 www.wiley.com/college/horstmann 在线获取。