



图灵计算机科学丛书

Apress®

Beginning Java Objects 中文版 从概念到代码 (第2版)

Beginning Java Objects: From Concepts to Code
Second Edition

[美] Jacquie Barker 著
万波 等译



人民邮电出版社
POSTS & TELECOM PRESS



图灵计算机科学丛书

Beginning Java Objects 中文版 从概念到代码 (第2版)

Beginning Java Objects: From Concepts to Code

Second Edition

[美] Jacquie Barker 著
万波 等译



人民邮电出版社
北京

图书在版编目 (CIP) 数据

Beginning Java Objects 中文版: 从概念到代码: 第2版 / (美) 巴克著; 万波等译.

—北京: 人民邮电出版社, 2007.1

(图灵计算机科学丛书)

ISBN 978-7-115-15440-8

I. J... II. ①巴...②万... III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 126172 号

内 容 提 要

本书是关于软件对象和 Java 的, 但并不是纯粹地介绍 Java 语言, 而是强调如何从对象模型转换到功能完整的 Java 应用程序。书中讲述了对象基础、对象建模和模型的实现。本书除了用学生注册系统 (SRS) 示例贯穿全书之外, 还在附录中给出三个附加的案例, 这些案例是每章结尾给出的诸多练习的基础。

本书适合作为高等院校计算机专业“面向对象程序设计”、“面向对象方法”或“Java 语言”课程的教材, 也适合作为公司的培训课程的教程。

图灵计算机科学丛书

Beginning Java Objects 中文版: 从概念到代码 (第2版)

◆ 著 [美] Jacquie Barker

译 万 波 等

责任编辑 杨海玲 黄 倩

◆ 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>

北京艺辉印刷有限公司印刷

新华书店总店北京发行所经销

◆ 开本: 787 × 1092 1/16

印张: 45.5

字数: 1 243 千字

2007 年 1 月第 1 版

印数: 1—5 000 册

2007 年 1 月北京第 1 次印刷

著作权合同登记号 图字: 01-2005-5700 号

ISBN 978-7-115-15440-8/TP · 5791

定价: 78.00 元

读者服务热线: (010)88593802 印装质量热线: (010)67129223

版 权 声 明

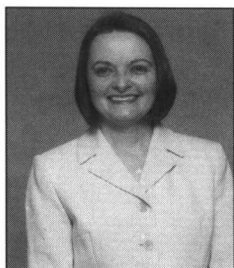
Original English language edition published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright © 2005 by Jacquie Barker. Simplified Chinese-language edition copyright © 2006 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Apress L.P.授权人民邮电出版社独家出版。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。

版权所有，侵权必究。

作者简介



JACQUIE BARKER 是一位专业的软件工程师、技术作家，也是位于弗吉尼亚州费尔法克斯市的乔治·梅森大学和位于华盛顿特区的乔治·华盛顿大学的兼职教员。在超过 28 年的实际软件开发和项目管理经历中，Jacquie 将过去 12 年的时间主要用在对象技术上，并成为了对象建模专家和 Sun 公司认证 Java 程序员。她是 ObjectStart LLC（一个指导和培训对象技术的公司）的创始人，她也是位于弗吉尼亚州赫恩登市的 Technology Associates 公司技术部的资深成员。

Jacquie 在位于俄亥俄州克利夫兰的凯斯西保留地大学获得计算机工程理学学士学位，并获得最高荣誉，在那里她被吸收为 Tau Beta Pi 国家工程荣誉协会会员。她在加州大学洛杉矶分校获得计算机科学理学硕士学位，主修软件系统工程，后来在乔治·梅森大学的信息技术专业进行研究生课程学习。

Jacquie 教授对象基础的成功方法一直受到全球读者的广泛赞扬，本书第 1 版被多所大学用作核心 IT 课程的主要教材。她的新书 *Taming the Technology Tidal Wave: Practical Career Advice for IT Professionals* 是一本关于如何维持技术领先的轻松指南，该书也同样得到了好评。

在个人方面，Jacquie 热爱的对象包括她的丈夫 Steve、宠物老鼠、小型长毛德国猎犬（也叫作 wiener 狗）、教学、写作和 Java 程序设计。在不从事计算机相关的事务时，Jacquie 和 Steve 喜欢骑摩托车旅行穿过弗吉尼亚郊区、骑双人自行车、在湖边享受安静的假期以及与亲朋好友共度时光。

要获得 Jacquie 已经出版的书籍和提供的服务的更多信息，请访问她的网站 <http://objectstart.com> 和 <http://techtidalwave.com>。

关于技术审校者

JAMES HUDDLESTON 是拥有 30 多年信息技术经验的独立顾问。他是 Apress 的 *Beginning C# Databases: From Novice to Professional* 一书的作者，并且曾经担任过许多关于各种计算机主题的书籍的技术审校或编辑。当他不编程或者翻译荷马史诗时，他喜欢帮助其他作者完善作品，而且更喜欢和他的三个孩子（Jared、Quinn 和 Tess）一起玩。

序

欢迎阅读本书！自从 2000 年 11 月第 1 版出版以来，我很高兴地收到了读者发来的很多邮件和好评，他们发现我的书是学习 Java 和面向对象编程的一个极好的“跳板”。

我的书以与编程语言版本无关的永恒准则为基础，这就意味着不必每次 Sun 公司发布 Java 新版本时都修订本书。但由于 Java2 平台标准版 (J2SE) 5.0 引入了一些重要的新特征，显著地增强了 Java 作为面向对象编程语言的能力和通用性，我认为这是以第 2 版的形式“更新”本书的恰当时机。

从第 1 版出版以来，我一直在以本书为素材进行教学，并且在教授许多大学和公司学生之后，显著地改善了书稿。和学习一样，教学是一项不断提高的工作，一个人的信息永远不可能 100% 的完整！

本版中值得关注的修改包括：

- 强调通过使用 Java 更快地获得实际经验（正如本书第 2 章与第 1 版的第 13 章的差别）。
- 在第 15 章对模型-数据层分离的讨论做出了重大的改进，包括对 JDBC API（用于和 ODBC 兼容的数据库通信）的概念性介绍。
- 第 17 章重点强调了模型-视图分离，包括对 J2EE 技术的概念性介绍。
- 强调了从 J2SE 5.0 版本开始对 Java 语言所做出的重大改进，并且以 OO 术语突出了它们的重要性。

在认识到本书已经广泛地被高等院校用作教材之后，我保持本版的章节大纲与第 1 版相同，唯一的不同之处在于：由于第 1 版的第 16 章在本版中显得过于庞大，我把它分成了两章。

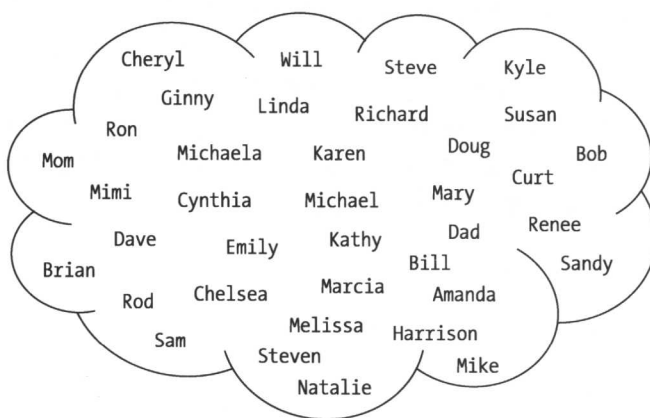
和以往一样，我欢迎读者的反馈并且希望通过我的两个网站尽快获得你们的意见：
<http://objectstart.com> 和 <http://techtidalwave.com>。

Jacquie

致 谢

我诚挚地感谢：

- ❑ James Huddleston, 我的技术审校。在整理本书的过程中他提供了经过深思熟虑的英明的指导。James, 你是一个真正的绅士, 很好合作 (更不用说还才华横溢了!), 我盼望将来更多地与你合作。
- ❑ Beckie Stones、Nicole LeClerc、Katie Stence 和 Steve Anglin。在完成本书时他们提供了极好的支持。没有你们就无法完成本书! (同样感谢 Paul Carlstroem 和 Stephanie Parker, 他们的市场经验使我的书特别成功。)
- ❑ Gary Cornell, Apress 的作者和出版人, 一直是本书的忠实读者和支持者。
- ❑ 我的朋友 Cathy McCabe、Brian Coleman 和 Cynthia Coleman。他们非正式地审读了本书并提供了关于如何改进书稿的无价建议。
- ❑ Claudio Cioffi 博士以及 George Mason 大学 2005 春季 Computational Social Sciences 605 班的学生, 他们帮助我进一步润色书稿使它适合更多的读者。
- ❑ 很多耐心等待我从写书的“社交深渊”返回的可爱的人, 他们是:



最重要的是感谢我生命中的爱人、我最好的朋友、我的丈夫 Steve, 他是给予我如此多的支持和理解的伴侣。我愿意和你共度一生, “Klemmie!”

前言

首先，这是一本关于软件对象的书：软件对象是什么，为什么它们具有如此大的“魔力”却又如此简单易懂，如何正确地使用对象来构建一个软件应用程序。

这也是一本关于 Java 的书。它不是一本绝对的、“一切为了了解 Java”的书，而是平和而全面地介绍 Java，特别强调如何从对象模型转换到功能完整的 Java 应用程序——这是其他书很少提供的内容。

本书的目标

在写作本书时我的目标（也希望你购买本书的原因）是：

- 使你熟悉基本的面向对象（OO）术语和概念。
- 让你亲自动手，实际体验对象建模，也就是说，开发一个以后可以用作构建面向对象软件系统基础的蓝图。
- 阐明这样的对象模型如何被转换为可以工作的软件应用程序——具体地说是 Java 应用程序，虽然你学到的对象建模技术同样适用于其他 OO 语言。
- 在这个过程中帮助你成为一个熟练的 Java 程序员。

如果你已经有 Java 语言编程的经验（但对对象基础没有经验），了解 Java 的面向对象的核心是成功使用这种语言的关键。另一方面，如果你是 Java 新手，本书将使你有一个正确的开始。不管怎样，这是任何希望精通 Java 这样的 OO 编程语言的人必读的一本书。

同样重要的是，这本书并不打算：

- 使你一夜之间成为对象建模的专家。和所有高级技巧一样，要完全掌握对象建模需要做到两件事：良好的理论基础和大量的实践。在本书中给出了理论基础，包括对统一建模语言（UML）的介绍，UML 是产生软件应用程序的面向对象“蓝图”的业界标准。但真正精通对象建模的唯一途径是参与到面向对象建模和项目开发之中。

本书将教给你技巧也希望能给你在专业背景中应用对象技术的信心，那才是你真正开始学习的地方，特别是如果有一个具有 OO 经验的指导者能够指导你完成你的第一个具有“业界水平”的项目。

- 教给你了解 Java 所需的一切知识。Java 是一种丰富的语言，它包含了上百个核心类以及差不多几千个在这些类上执行和由这些类执行的操作。如果 Java 提供了解决某个特定问题的多种备用方案，我会解释其中一到两个最适合解决这个问题方法以使你理解问题应该如何解决。尽管如此，在本书中你仍然会看到足够多的 Java 语言以帮助你成为职业的 Java 程序员。

有了本书学到的知识以后，你可以泰然自若地理解现有的其他 Java 参考书对 Java 的处理，或者以彻底的 UML 标准来更加深入地回顾对象建模技术。

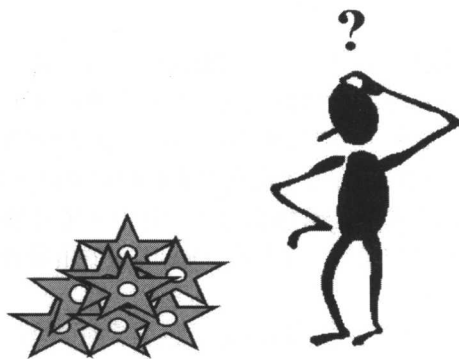
为什么对对象的理解是成为成功 OO 程序员的关键

在我就职的地方、客户办公室、专业会议以及大学校园，我多次遇到这样的软件开发人员，他们试图通过学习 Java 课程、阅读关于 Java 的书籍或者安装并使用 JBuilder 或 Sun One Studio 这样的 Java 集成开发环境（IDE）来掌握 Java 这样的面向对象编程语言。然而，实际上所有的这些方法都从根本上缺少一些东西，即对对象的基本理解，并且更重要的是如何最大程度地利用对象从基础开始构建软件应用程序的知识。

假设你被要求建造一所房屋，并且你知道构建房屋的基本知识。事实上，你是一个世界知名的建筑师，你提供的服务有很大的需求。你采用过各种可能的建筑风格，使用过各种已知的建筑材料：砖、木材、石头、金属等。因此，当你的客户要求你使用一种他提供的新型材料时，你愉快地答应了。



建造开始的那一天，一辆卡车在建筑地点停下并卸下一大堆奇形怪状、中间有孔的蓝色星形材料。你完全迷惑了！你已经用多种熟悉的材料建造了无数房屋，但对如何使用这种蓝色星形材料建造一所房屋毫无头绪。



冥思苦想之后，你拿出一把铁锤和一些钉子并试图把这些蓝色星形材料像钉木材一样钉在一起，但这些星形材料组合得不好。然后，你试图用黏合砖块的灰泥来填充它们的空隙，但灰泥不能很好地和这些蓝色星形材料黏合在一起。然而，由于你为客户建造房屋的经费和时间都很紧张（而且作为一个建筑设计专家，你不允许自己不知道如何使用这种现代材料），你只能进行下去。最后，你还是造出了一个（至少在表面上）看起来像房屋的东西。



你的客户来检查工作，他非常失望。他选择这种蓝色星形建筑材料的一个原因是这些材料非常节能，但由于你使用钉子和灰泥来装配这些星形材料，它们丢失了大部分固有的保温能力。

为了补救，你的客户要求你为房屋的所有窗户换上三层保温玻璃，以减少热量散失。你对这个要求感到恐慌——因为换窗户需要把墙拆开，会破坏房屋。

当你把这一点告知客户时，他非常生气！他选择这种蓝色星形建筑材料的另一个原因是由于这些材料的模块性因而很容易适应设计改变，但由于你用无效的方式来装配这些材料，它们也失去了灵活性。



遗憾地说，这是很多程序员在没有接受应用程序构建单元（即软件对象）的基本特性培训的情况下最终构建 OO 应用程序的方式。更糟糕的是，绝大多数想要成为 OO 程序员的人都欣然忽视了为了使用 OO 语言编程而理解对象的需要。他们匆忙地使用 Java 这样的语言编程并最终得到完全不理想的结果：一个在响应需求改变方面缺少灵活性的应用程序。

本书面向的读者

希望最大程度地利用 Java 这样的面向对象编程语言的任何人！本书适合于：

- ☐ 已经从事 Java 工作，但希望有一个正确的开始的人。
- ☐ 已经购买了关于 Java 的书（并虔诚地阅读了这些书），懂得了这种语言的技术问题，但还不十分清楚如何最好地利用 Java 的面向对象特征来构建应用程序的人。
- ☐ 已经使用过 Java IDE，但只知道如何拖放 GUI（图形用户界面）组件以及如何给按钮、

菜单等组件添加后台逻辑，而不知道如何围绕这些对象正确地构建应用程序核心的人。

- ❑ 已经构建过 Java 应用程序，但对根据应用程序生命周期中提出的新需求进行维护和修改的困难性感到失望的人。
- ❑ 以前学过一些对象建模知识，但对如何将对象模型转变为真实代码（Java 代码或其他语言的代码）感到迷惑的人。

底线是，任何希望精通一门 Java 这样的 OO 语言的人必须首先成为一个对象领域的专家！

为了从本书获得最大价值，你应该具有一些编程经验，实际上任何语言的编程经验都可以。

你应该理解简单的编程概念，例如：

- ❑ 简单的数据类型（整型、浮点型等）。
- ❑ 变量和作用域（包括全局数据的概念）。
- ❑ 流程控制（if...then...else 语句、for/do/while 循环等）。
- ❑ 什么是数组以及如何使用数组。
- ❑ 软件函数/子程序/过程的概念：如何传入数据并返回结果。

但你不必接触过 Java，也不必接触过对象——至少软件意义上的对象你不必接触过！从第 1 章你会了解到，人类会很自然地从一个对象的角度来观察整个世界。

即使你已经开发过完整的 Java 应用程序，但如果你在面对构建应用程序的对象特征时仍然感到迷惑，阅读本书无疑不算晚：它最终会使你成为更好的 Java 程序员并了解面向对象的“原理”而不仅仅是语言的机制。在本书中你极有可能看到一些熟悉的概念（以 Java 示例代码的形式呈现），但你在学这些基本原理时有希望获得许多关于为什么在使用 Java（或者任何其他 OO 编程语言）编程时要这样做的新见解。

由于本书的根源是我在大学讲授的课程，它很适合作为大学的一个学期的 Java 编程或对象建模课程的教材。在附录 A 中包含了将本书用作教材的一些建议。

如果你对对象建模感兴趣但不用从事 Java 编程呢

这本书对你还有价值吗？毫无疑问！即使你不打算在编程上有所成就（正如我的学习对象建模的学生一样），我发现了解用 Java 这样的 OO 语言编写的示例代码确实有助于巩固对象概念。因此，鼓励你阅读第三部分，这一部分逐步建立了一个完整的 Java 应用程序，即使你从来没有打算在以后进行 Java 编程。

本书的组织方式

本书是围绕三个主题来构建的，如下所述。

第一部分：对象基础

在钻研对象建模入门指导和 Java OO 编程细节之前，统一对象方面的语言是非常重要的。第一部分由第 1 章到第 7 章组成，慢慢地从定义支持所有软件开发方法（OO 方法或者其他方法）的基本概念开始。但这些章节迅速地进入到对高级对象概念的讨论中，因此，到第一部分结束时，你应该“了解对象”了。

第二部分：对象建模基础

在第二部分（本书第 8 章到第 12 章）强调开发应用程序的对象模型时应该怎么做以及为什么要这样做的基本准则，这些准则对所有对象建模技术是通用的。熟悉 UML 是很重要的，因此这一部分教给你 UML 基础并将它用于本书中所有具体的建模示例。使用在这些章节给出的建模技术，可以为学生注册系统（SRS）开发一个对象模型“蓝图”，这个系统的需求说明在“引言”的最后给出。

第三部分：将对象蓝图转换为 Java 代码

第三部分（本书第 13 章到第 17 章）阐明了如何将第二部分开发的 SRS 对象模型变成功能完整的 Java 应用程序，用 GUI 和一种使不同的用户登录之间的数据持久化的方法来完善整个应用程序。在这一阶段给出的所有示例代码都可以从 Apress 站点 (<http://www.apress.com/>) 下载，我强烈推荐你下载并实验这些代码。

SRS 的需求说明用叙述性风格写成，这是软件系统表示需求通常采用的方式。现在你可能对你能够构建一个应用程序来解决这个问题感到自信，但到本书结束时你应该为能够构建一个面向对象应用程序来解决这个问题而更加自信。附录 B 中给出 3 个附加的案例研究（处方跟踪系统、会议室预定系统和机票预定系统），这些案例是每章结尾给出的诸多练习的基础。

为了使本书更加完善，我还引入了第 18 章，对读完本书后应该如何继续面向对象的探索进程给出了一些建议。在这一章中提供了一个推荐书目的列表，这些书将把你带到更加精通的层次，这个层次取决于你打算将本书中学到的知识应用于什么场合。

约定

为了帮助你最大程度地利用本书并了解正在进行的是什么，本书采用了许多约定，例如：

注解以这种形式呈现，并反映了重要的背景信息。

以下是你在书中会发现的风格：

- 当引入重要的词语时，我会用**黑体**突出显示它们。
- 文件名、URL 和代码以这样的文本显示：`objectstart.com`。
- 在长代码段中希望引起你注意的代码行用**粗体**显示，例如：

```
// Bolding is used to call attention to new or significant code:  
Student s = new Student();  
// whereas unbolded code is code that's less important in the  
// present context, or has been seen before.  
int x = 3;
```

- 用斜体而不是正体来表示伪代码：

```
// This is real code:  
for (int i = 0; i <= 10; i++) {  
    // This is pseudocode!  
    compute the grade for the ith Student  
}
```

本书基于哪一个版本的 Java

和任何其他编程语言一样，Sun 公司时常会发行新版本的 Java。好消息是在本书中只强调核心 Java 语法（即从 Java 出现以来一直稳定的语言特征），本书和具体版本无关。你在本书中学到的技术和概念对新版本的 Java 同样适用。

Sun 公司在 J2SE 5.0（在 Sun 公司内部称为 Java 1.5.0）引入了一些重要的新特征，这些新特征显著地增强了 Java 作为一种面向对象编程语言的能力和通用性。在本书中已经采用了 J2SE 5.0，这影响了第 6 章以及第 13 章到第 16 章的示例代码。对那些仍然使用早期的 Java 版本工作的读者，我在这些章节包含了注解以指导你在需要时修改代码来适应早期的 Java 版本。

Java 术语

自从 1995 年 Java 第一次出现以来，Java 语言在版本的称谓上有一些不连续。

Java 的第一个版本称为 Java 1.0，其后是 Java 1.1。然而，当 Java 1.2 出现后，Java 语言在如何处理 GUI 事件方面有了显著的区别，以至于 Sun 公司把这个版本指定为“Java 2”（但仍然可以称为 1.2.x），其后的 1.3.x 和 1.4.x 版本称为“Java 2 平台 1.x.x”。然而当 Java 1.5.0 发布后，它再次整合的显著区别足以使它被重新命名为 J2SE 5.0（在第 6 章和第 13 章以及附录 G 中讨论了 J2SE 5.0 的一些重要改进）。

表 0-1 总结了到 Java 2 平台 1.5.0 为止的命名历史。

表 0-1 Java 命名历史*

实际版本号	建议名称
1.0	Java 语言 1.0
1.1.x	Java 语言 1.1
1.2.x	Java 2 平台
1.3.x	Java 2 平台 1.3.x
1.4.x	Java 2 平台 1.4.x
1.5.0 (现称 5.0.x)	Java 2 平台 1.5.x (现称 Java 2 平台 5.0.x)

*这个表中的版本号只适用于 J2SE 和 J2EE。

Java 开发平台 (JDK) 的称谓也存在类似的不连续性。JDK 是开发人员用来构建 Java 应用程序的软件“包”，它包括：

- Java 虚拟机 (JVM)。
- Java 编译器 (javac)。
- Java 归档 (jar) 实用程序。
- Java 文档 (javadoc) 实用程序。

等等。命名出现不连续性的部分原因是 Java 语言的 1.2 版本出现了三种 Java “平台”：

- Java 2 平台微型版 (J2ME)：Java 的精简版非常适合于消费类电子设备编程，例如移动电话、寻呼机、个人数字助手 (PDA) 等。
- Java 2 平台标准版 (J2SE)：基本 (最初) 的 Java 语言和运行环境，用于构建桌面应用程序，例如在本书第三部分将要构建的 SRS。J2SE 也是 Java 企业版 (J2EE) 的基础。
- Java 2 平台企业版 (J2EE)：和 J2SE 一起用于构建多层企业级应用程序的组件技术

集，在第 17 章将对它进行讨论。

因此，Java 开发人员需要的软件包可以被称为以下任何一种：

- ☐ Java 开发包 (JDK)。
- ☐ (Java) 软件开发包 (SDK)。
- ☐ Java 2 软件开发包 (Java 2 SDK)。
- ☐ Java 2 平台标准版 (J2SE) 开发包。

本书将这个软件包称为 JDK 或 SDK。

最后，还有 Java 运行环境 (JRE) 的概念，JRE 是 JDK 的子集，包括 JVM、运行时类库和执行 Java 字节码所需的 Java 应用程序启动器，但省略了 Java 编译器 (javac) 这样的开发工具。JDK 的每一个发行版本号 1.x.x 都有一个对应的 JRE 发行号。

开始学习之前的最后一个问题

本书中的许多材料（特别是在第一部分的开始部分）在富有经验的程序员看来过于简单，这是因为很多对象技术建立在基本的软件工程准则之上，这些准则已经在实践和很多案例中应用了多年，只是包装得稍有不同。但确实有些新技术使得 OO 语言极其强大，而且这些技术确实不能用非 OO 的语言实现，例如继承和多态性，在第 5 章和第 7 章会分别了解到和它们有关的更多知识。（这些技术可以在非 OO 语言中手工模拟，这就像程序员可以从零开始编写自己的数据库管理系统 (DBMS) 而不是使用 Oracle、Sybase 或 Microsoft SQL Server 这样的商业产品一样——但是谁愿意这样做呢？）

在精通对象的过程中，经验丰富的程序员面临的最大问题是重新适应思考他们要自动化的问题的方式。

- ☐ 已经使用非面向对象的方法开发过应用程序的软件工程师/程序员通常必须“忘掉”某些在传统的软件分析和设计方法中使用的方式。
- ☐ 相反的是，那些把面向对象的软件开发方法作为他们唯一的方法进行学习的初级程序员有时更加容易。

幸运的是，我们在开发软件时思考对象的方式正是人们考虑一般世界的自然方式。因此，按本书的三个部分学习“思考”对象（并且用 Java 语言实现它们）就像学习 1、2、3 一样容易！

告诉我你的想法

我和 Apress 的编辑以及出版职员努力地工作以使本书对你尽可能有用，我们也很乐意知道你的想法！我们重视读者的反馈，并接受建设性的批评和赞扬以便更好地进行将来的工作。

如果你想和 Apress 的任何人联系，请发邮件到 info@apress.com 或者登录 <http://www.apress.com>。你可以通过 jacquie@objectstart.com 或者我的网站 <http://objectstart.com> 和 <http://techtidalwave.com> 和我联系。

学生注册系统 (SRS) 案例研究： SRS 需求说明

要求开发一个自动的学生注册系统 (SRS)，允许学生在线注册每学期的课程并记录学位完成的进度。

当学生第一次入学时，该学生使用 SRS 提出一个学习计划，即，为了满足特定学位程序的要求而计划学习的课程，并选择一个指导教师。SRS 会验证这个学习计划是否满足学生攻读的学位的要求。一旦建立了一个学习计划，在每学期开始前的注册期间，学生都能在线查看课程计划，并选择要上的课程，如果多个教授开设这门课程，学生还可以指明听课时间（每周哪一天和每天的什么时候）。SRS 通过在线查阅学生已完成的课程和获得的成绩来验证学生是否满足请求的每一门课程的先修科目（学生可以随时在线查看自己的成绩单）。

假设(a)满足请求的课程先修科目，(b)请求的课程满足学生的学习计划要求，并且(c)每门课程都有空余的选课名额，这个学生就成功注册。

如果(a)和(b)满足但(c)不满足，学生被放入先来先服务的等待队列。如果学生先前等待注册的课程变为可选（可能是因为某些学生取消了这门课程或者这门课程的课堂座位容量增加了），这个学生将被自动注册到等待注册的课程，并且收到确认注册有效的电子邮件消息。如果学生不再需要这门课程，那么必须取消这门课程，否则该学生必须为这门课程付费。

学生最晚可以在开设该课程的学年的第一周周末之前取消已注册的课程。

目 录

第一部分 对象基础

第 1 章 抽象和建模	2
1.1 通过抽象来简化	2
1.2 通过抽象来概括	3
1.2.1 将抽象组织成类层次结构	3
1.2.2 将抽象作为软件开发的基础	6
1.3 抽象的重用	6
1.4 固有的问题	7
1.5 小结	9
第 2 章 Java 基础	10
2.1 为什么选择 Java	10
2.1.1 Java 是体系结构中立的	10
2.1.2 Java 提供“一站式服务”	13
2.1.3 Java 是完全面向对象的	15
2.1.4 实践使 Java 更完美	16
2.1.5 Java 是一种开放标准	16
2.1.6 Java 是免费的	16
2.1.7 关于伪代码和真正 Java 代码的提醒	16
2.2 简单 Java 程序剖析	17
2.2.1 注释	17
2.2.2 类声明	18
2.2.3 main 方法	19
2.3 Java 的“机制”	19
2.3.1 把 Java 源代码编译成字节码	21
2.3.2 执行字节码	21
2.3.3 JVM 内幕	22
2.4 基本类型	22
2.5 变量	23
2.6 变量初始化	25
2.7 String 类型	25
2.8 区分大小写	26
2.9 Java 表达式	27
2.9.1 算术运算符	27
2.9.2 关系和逻辑运算符	28
2.9.3 表达式求值和运算符优先级	29
2.9.4 表达式类型	30
2.10 自动类型转换和显式转型	30
2.11 循环和其他流程控制结构	32
2.11.1 if 语句	32
2.11.2 switch 语句	34
2.11.3 for 语句	35
2.11.4 while 语句	37
2.11.5 跳转语句	38
2.12 块结构语言和变量作用域	39
2.13 打印到屏幕	40
2.13.1 print 与 println	41
2.13.2 转义序列	42
2.14 Java 风格要素	43
2.14.1 适当使用缩进	43
2.14.2 明智地使用注释	46
2.14.3 大括号位置	46
2.14.4 描述性的变量名称	47
2.15 小结	47
第 3 章 对象和类	49
3.1 最简软件	49
3.1.1 功能分解	49
3.1.2 面向对象方法	51
3.2 对象是什么	51
3.2.1 状态/数据/属性	52
3.2.2 行为/操作/方法	53
3.3 类是什么	53
3.3.1 关于命名约定的提示	54