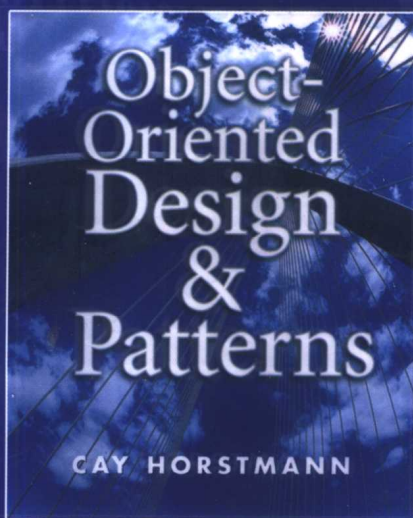


国外计算机科学教材系列

面向对象的设计与模式

Object-Oriented Design & Patterns



[美] Cay Horstmann 著

张琛恩 译



电子工业出版社

Publishing House of Electronics Industry
<http://www.phei.com.cn>

国外计算机科学教材系列

面向对象的设计与模式

Object-Oriented Design & Patterns

[美] Cay Horstmann 著

张琛恩 译

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书通过讲授Java语言,将面向对象的设计原理和设计模式相互结合起来。本书的重点是从设计的角度出发,循序渐进地向读者全面介绍了面向对象的设计(方法学,CRC卡和UML类图,设计模式),用户接口编程(Swing应用程序,事件处理,Java 2D图形),Java语言的高级特性(接口类型,内部类,反射,多线程技术),以及一些常用开发工具的使用。另外,本书还配有大量完整的程序代码并包含了详细的注释,可供任何面向对象编程技术的初学者或者是具有一定基础的读者参考和使用。

Cay Horstmann: Object-Oriented Design & Patterns

ISBN 0-471-31966-X. Copyright © 2004 John Wiley & Sons, Inc. All rights reserved.

Authorized translation from the English language edition published by John Wiley & Sons, Inc.

No part of this book may be reproduced in any form without the written permission of John Wiley & Sons, Inc.

Simplified Chinese translation edition published by Publishing House of Electronics Industry. Copyright © 2004.

本书中文简体字翻译版由 John Wiley & Sons 授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

版权贸易合同登记号 图字:01-2003-3680

图书在版编目(CIP)数据

面向对象的设计与模式/(美)霍斯特曼(Horstmann, C.)著;张琛恩译.—北京:电子工业出版社,2004.1
(国外计算机科学教材系列)

书名原文:Object-Oriented Design & Patterns

ISBN 7-5053-9575-0

I.面... II.①霍... ②张... III.面向对象语言-程序设计-教材 IV.TP312

中国版本图书馆CIP数据核字(2003)第125746号

责任编辑:李秦华 熊 健

印 刷:北京兴华印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

经 销:各地新华书店

开 本:787×980 1/16 印张:25.75 字数:546千字

印 次:2004年1月第1次印刷

定 价:35.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换;若书店售缺,请与本社发行部联系。联系电话:(010)68279077。质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出版说明

21世纪初的5至10年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入WTO后的今天,培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择 and 自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如Pearson Education培生教育出版集团、麦格劳-希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过与作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

主 任	杨芙清	北京大学教授 中国科学院院士 北京大学信息与工程学部主任 北京大学软件工程研究所所长
委 员	王 珊	中国人民大学信息学院院长、教授
	胡道元	清华大学计算机科学与技术系教授 国际信息处理联合会通信系统中国代表
	钟玉琢	清华大学计算机科学与技术系教授 中国计算机学会多媒体专业委员会主任
	谢希仁	中国人民解放军理工大学教授 全军网络技术研究中心主任、博士生导师
	尤晋元	上海交通大学计算机科学与工程系教授 上海分布计算技术中心主任
	施伯乐	上海国际数据库研究中心主任、复旦大学教授 中国计算机学会常务理事、上海市计算机学会理事长
	邹 鹏	国防科学技术大学计算机学院教授、博士生导师 教育部计算机基础课程教学指导委员会副主任委员
	张昆藏	青岛大学信息工程学院教授

译者序

一段时间以来，我在讲授 Java 语言、数据结构、软件工程等这些与面向对象设计相关的课程中，学生的反馈使我深深体会到，有时确实很难将这些知识很好地融合起来，尽管它们都属于面向对象设计中的重要内容。但这些孤立的课程，要么强调的是某种语言的具体语法，要么强调的是某些具体的算法，要么就是具体的使用技术或者是过于理论化的知识。如果没有真正理解面向对象的思想，将很难开发出更为规范的软件，继而难以激发起他们继续深入学习的动力。

本书能够让初学者在较短的时间内迅速掌握面向对象的设计知识，使他们站在一个较高的层面上来看待有关面向对象的知识，解决实际设计过程中所遇到的问题，并利用 Java 语言将设计原理同设计模式相互结合。

本书利用 Java 这门流行的面向对象语言，将面向对象的设计原理同设计模式结合在一起，并配有大量的、通过 Swing 图形用户接口工具设计的实例，使读者能够迅速掌握 Java 语言的基础知识及相关的高级应用，使读者从设计模式的角度去解决相关的设计问题，从而顺利进入更高级的课程学习之中。通过对一些经典模式的学习，相信能对面向对象技术的初学者或者是具有一定基础的开发人员提供相应的帮助。

由于本书采用了最新的 Java SDK 1.4 版本中的技术，在翻译过程中，为了能够使读者更好地理解书中的内容，个别术语在采纳过程中与一些同学进行了交流，一方面要忠实于原文中的内容，另一方面是尽量能从读者易于理解的角度来传达作者的意思。这里，我想衷心感谢曾给过我帮助的长安大学信息学院的广大师生们，以及西安交大金科国际认证培训中心的师生们，正是因为他们的支持和鼓励，使得本书最终能与读者见面。但由于译者本身的知识水平有限，书中难免有一些不当之处，恳请广大读者批评指正。

前 言

了解面向对象的设计

在计算机科学的学习过程中，一定会使用到面向对象的设计原理。本书的目的在于：在早期的课程中讨论这些设计原理。

ACM/IEEE 2001 课程指南针对计算机科学课程的设置顺序定义了一种“对象优先”的概念。计算机课程设置顺序中的第一门课程通常是没有任何编程背景的学生介绍一门面向对象的编程语言（通常是 Java）。学生们将学习类、方法、控制流程、输入和输出、数组和一些继承结构。当然，对于面向对象的设计概念，不可能在第一门课程中就获得深刻的理解。这样，需要一门独立的面向对象的设计课程来介绍这些概念。

一种可能的解决办法是将数据结构与面向对象的设计原理相结合。然而，这种方法可能会产生问题。数据结构和算法的一些经典主题很可能在这门课程中显得要比面向对象知识更重要。此外，学生们通常对数据结构的设计缺乏热情，并且难以将这些主题与面向对象的设计原理联系起来。

本书采用了一种完全不同的方法。它把图形用户接口的编程技术（一个令学生很感兴趣的话题）运用到了设计模式和面向对象的设计原理之中。

为进一步学习奠定基础

在学习完本书以后，学生应该能够掌握以下三个领域中的主题：

1. 面向对象的设计

- 一种简单的设计方法
- CRC 卡与 UML 图
- 设计模式

2. 图形用户接口设计

- Swing 应用程序
- 事件处理机制
- Java 2D 图形

3. 高级 Java 语言

- 接口类型
- 内部类
- 反射机制
- 多线程

掌握这些技术会为日后计算机科学的高级课程打下一个坚实的基础。实际上，学生们在学习完这本书以后，将遇到 Java 语言的所有特征（当然，不会是整个 Java 类库，其庞大的内容对任何人来讲都是很难掌握的）。使用 Java 的一个优势是：学生能够真正掌握整个语言。C++则的确过于复杂，几乎没有人敢真正宣称自己理解了 C++语言的所有细节问题。

如果希望学生在较早的课程中理解面向对象的设计与设计模式，可以使用这本书。学生们将学会使用 Swing 用户接口编程，并对 Java 语言有一个更全面的理解。

与设计模式相融合

本书最显著的特点是将设计模式与某些部分融合在一起。例如：

- Swing 容器与组件的使用产生了 COMPOSITE 模式
- Swing 滚动条的使用产生了 DECORATOR 模式，该模式同时为 Swing 的边框技术提供了一种参考
- Java 流技术是 DECORATOR 模式的另一个例子。这种不同的表现方式能够清楚地阐明这一模式的概念

编程与设计工具

本书的另外一个特点是采用了一些有用的工具。尽管 C++程序员的整个编程阶段都是在一种大规模的、复杂的集成环境中完成的，Java 系统同样也有许多不同的工具，如 BlueJ, javadoc 和 JUnit。由于能够反映 Java 语言的本质，本书还将介绍一些有趣的实验工具并希望学生们可以从中发现乐趣，从而能够在未来的工作环境中掌握这些开发工具。

学习面向对象设计的学生们还应该熟悉绘制 UML 图。一种易于使用并且免费的工具——Violet UML 编辑器，可以帮助完成这一目的。在本书第 8 章中，将介绍采用 Violet 作为基础的框架技术。此外，本书中的所有 UML 图都是使用 Violet 绘制的。

本书导读

第 1 章 Java 简介

本章介绍基本的 Java 语法，它也可以作为一种回顾性的复习资料，或者是用做具有

C++背景的学生的过渡材料。本章主要包含以下主题：

- 定义类及方法
- 对象及对象的引用
- 利用 BlueJ 探究对象
- 文档注释
- 数位、字符串和数组
- 包
- 异常处理机制
- 常用工具类：ArrayList、StringTokenizer 和 JOptionPane（用于输入）
- 编程风格

第 2 章 面向对象的设计过程

本章介绍面向对象的设计过程、CRC 卡以及 UML 标记。其中给出了一个简单的语音邮件系统，举例说明了设计的整个过程并以方案规格说明书作为开始，最终实现该系统。本章主要包含以下主题：

- 识别类与方法
- 类之间的关系
- CRC 卡
- UML 类图、时序图和状态图
- 案例研究

第 3 章 类设计的指导方针

不同于第 2 章中采用的自顶向下的设计方法，本章强调的是对一个类或一组相关类的设计。本章主要包含以下主题：

- 设计与实现一个类的接口
- 封装的重要性
- 接口质量分析
- 合同式编程：先决条件、结果状态和恒定式

第 4 章 接口类型与多态

本章在未涉及到继承的概念时，首先介绍了 Java 接口类型的表示方法。这种方式有一个重要的优点：读者将以简单的形式学习多态技术，因而不会被超类或超类方法调用这些

技术性问题所干扰。

本章还将介绍 Swing 用户接口工具包和 AWT 图形操作。首先介绍了 Icon 接口，该接口允许在一个图文框中放置任意的图形。

匿名类是一种为特殊对象而实现特定接口的便利机制。它们将应用在 Swing 的用户接口驱动中。

目前为止，所有的接口类型都是由标准类库提供的。然后，本章将以用户自定义接口类型的设计作为结束。本章包含以下主题：

- 图文框、图像和图形
- Icon 接口
- Comparable 接口和 Comparator 接口
- 匿名类
- 用户接口驱动
- 接口类型设计

第 5 章 模式与 GUI 程序设计

本章将介绍模式的概念，以及在 Swing 用户接口工具包和 Java 类库中出现的一些模式。本章包含以下主题：

- Alexander 结构模式
- 软件设计模式
- 设计模式的一个范例——ITERATOR 模式
- OBSERVER 模式、模型/视图/控制器及 Swing 监听程序
- STRATEGY 模式和布局管理
- COMPOSITE 模式、用户接口组件和容器
- DECORATOR 模式、滚动窗体和边框

第 6 章 继承和抽象类

本章通过 AWT 图形库中的实例，向读者介绍了继承机制。同时，还将讨论抽象类这一主题并使用一个抽象的图形类，作为第 8 章中所创建的图形编辑器的基础。本章还将考察一些采用继承的类集，包括 Swing 组件的类集结构、几何图形的类集结构和异常的类集结构。本章还将讨论高级异常处理机制，包括新建异常类的定义（当然要使用到继承）。最后，将讨论何时不使用继承。本章包含以下主题：

- 定义并实现子类

- 调用超类的构造方法和超类的方法
- 抽象类和 TEMPLATE METHOD 模式
- Swing 组件的继承结构
- java.awt.geom 包中图形类的继承结构
- 何时不使用继承

第 7 章 Java 对象模型

本章详细介绍了 Java 的类型系统以及 Java 的反射机制。然后讨论 Object 类中一些严格定义的基础方法：toString、equals、hashCode 和 clone。作为采用反射技术的一个应用程序，本章提出了 JavaBean 这一概念。本章最后的属性检查工具会在第 8 章的图形编辑器框架中再一次使用到。本章包含以下主题：

- Java 的类型系统、基本类型、包装类和数组类型
- 类型询问与反射机制
- 对象的等价性与克隆
- 串行化
- 组件和 JavaBean
- JavaBean 的属性

第 8 章 框架技术

本章将讨论图形编辑器框架的开发及其所专用的 UML 类图编辑器。本章包含以下主题：

- 框架概念
- 一个简单框架——applet
- 集合类框架
- 应用程序框架
- 图形编辑器框架

第 9 章 多线程

本章介绍基本的 Java 多线程技术，至此，将学习完 Java 语言所有的基础知识点。本章包含以下主题：

- 线程与 Runnable 接口
- 中断线程
- 线程同步
- 应用程序：算法的直观显示

第 10 章 更多模式

本章将讨论其他一些经典的设计模式——一些已经超出本书范围的经典设计模式。本章包含以下主题：

- ADAPTER 模式
- 动作与 COMMAND 模式
- FACTORY METHOD 模式
- PROXY 模式
- SINGLETON 模式
- VISITOR 模式
- 其他设计模式

图 1 是这些章节之间的关系依赖图。

教学结构安排

每一章都以引言和列有本章要点的清单作为开始。概念和原理会出现在编程实例的上下文中，许多实例程序也将完整地打印出来，以鼓励学生阅读并理解这些程序代码。

所有章节都包含一些有助于学生理解的注释。这些注释由专门的符号表示，因此不会打断正文内容的连续性。

专题中的内容是可以跳过的背景材料或者是较高级的主题。一些读者可能没有学习过数据结构，而这些相关内容将会出现在专题中，便于使用时参考。专题中的其余论题，还谈到了一些除 Java 之外与本章的设计原理相关的语言特征。

每一种设计模式的出现均采用一种标准的格式，包括该模式适用的场合、所提供的解决方案以及与之相关的 UML 图。另外，大部分模式后面均附有一张表，用于指导学生如何将该模式结构应用到实例中。

Web 资源

另外的一部分资料可以从本书的 Web 网站 <http://www.wiley.com/college/horstmann> 上进行查询。

这些资料包括：

- 编译器的帮助文档
- 用做讲义的幻灯片
- 本书所有实例的源代码

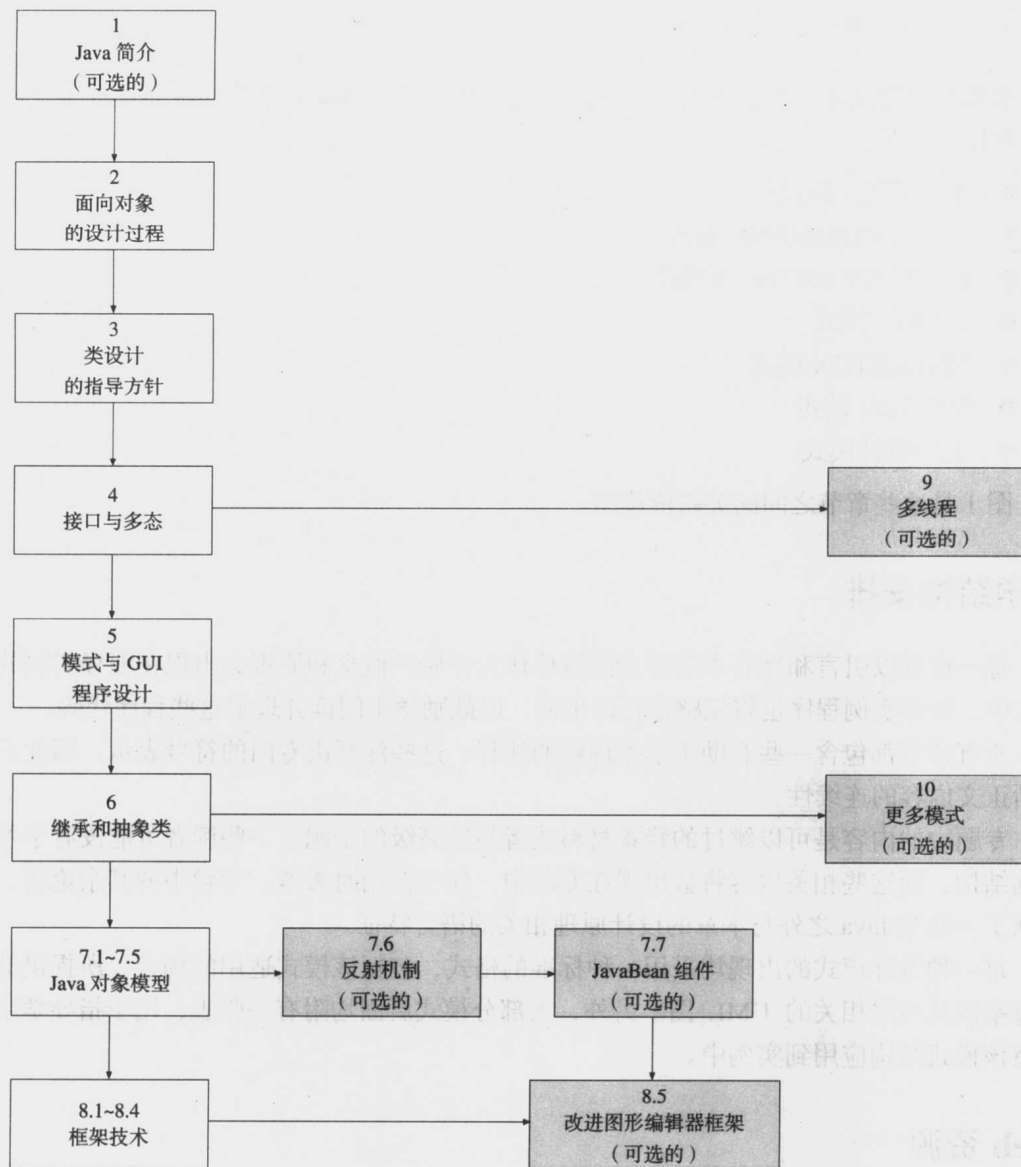


图 1 章节之间的关系依赖图

Violet UML 编辑器

Violet 是一个易于使用并且免费的 UML 编辑器，可以到 <http://horstmann.com/violet> 进行下载。正文中的 UML 图均由 Violet 绘制，因此，学生们也可以在设计程序时使用这一工具。

目 录

第 1 章 Java 简介	1
本章要点	1
1.1 利用 Java 实现 “Hello World!” 程序	1
1.2 文档注释	5
1.3 基本类型	8
1.4 控制流程语句	10
1.5 对象的引用	11
1.6 参数传递	12
1.7 包	15
1.8 基本异常处理	16
1.9 字符串	19
1.10 读输入	20
1.11 数组列表及链表	22
1.12 数组	24
1.13 static 属性及方法	25
1.14 编程风格	27
习题	29
第 2 章 面向对象的设计过程	32
本章要点	32
2.1 从问题到程序代码	32
2.2 对象与类的概念	35
2.3 识别类	37
2.4 识别职责	41
2.5 类之间的关系	42
2.6 用例分析	44
2.7 CRC 卡	46
2.8 UML 类图	48

2.9 时序图	52
2.10 状态图	54
2.11 利用 javadoc 设计文档	55
2.12 案例研究：一个语音邮件系统	56
习题	78
第 3 章 类设计的指导方针	81
本章要点	81
3.1 考察 Java 类库中的 Date 类	81
3.2 设计 Day 类	85
3.3 Day 类的三种实现方式	88
3.4 封装的重要性	99
3.5 接口质量分析	107
3.6 合同式编程	111
3.7 单元测试	122
习题	123
第 4 章 接口类型与多态	127
本章要点	127
4.1 Icon 接口	127
4.2 多态	132
4.3 图形绘制	133
4.4 Comparable 接口	137
4.5 Comparator 接口	140
4.6 匿名类	141
4.7 图文框与用户接口组件	144
4.8 用户接口驱动	147
4.9 定时器程序	150
4.10 接口类型设计	152
习题	158
第 5 章 模式与 GUI 程序设计	161
本章要点	161
5.1 迭代器 (Iterator)	161
5.2 模式的概念	163

5.3	OBSERVER 模式	166
5.4	布局管理与 STRATEGY 模式	169
5.5	组件、容器与 COMPOSITE 模式	180
5.6	滚动条与 DECORATOR 模式	181
5.7	如何识别模式	184
5.8	应用模式	186
	习题	198
第 6 章	继承和抽象类	200
	本章要点	200
6.1	继承的概念	200
6.2	利用继承实现图形编程	207
6.3	抽象类	213
6.4	TEMPLATE METHOD 模式	221
6.5	拥有保护的接口	224
6.6	Swing 组件的类集结构	227
6.7	标准几何图形的类集结构	231
6.8	Exception 的类集结构	236
6.9	何时不用继承	238
	习题	241
第 7 章	Java 对象模型	244
	本章要点	244
7.1	Java 的类型系统	244
7.2	类型查询	251
7.3	Object 类	254
7.4	浅复制与深复制	263
7.5	序列化	267
7.6	反射机制	270
7.7	JavaBean 组件	277
	习题	297
第 8 章	框架技术	300
	本章要点	300
8.1	框架	300

8.2 一个简单的框架——applet	301
8.3 集合类框架	304
8.4 一个图形编辑器框架	312
8.5 改进的图形编辑器框架	332
习题	336
第 9 章 多线程	338
本章要点	338
9.1 线程的基本知识	338
9.2 线程同步	346
9.3 线程与动画	359
习题	366
第 10 章 更多模式	368
本章要点	368
10.1 ADAPTER 模式	368
10.2 动作类与 COMMAND 模式	371
10.3 FACTORY METHOD 模式	376
10.4 PROXY 模式	378
10.5 SINGLETON 模式	382
10.6 VISITOR 模式	384
10.7 其他设计模式	391
习题	393