Java程序员书库

Java设计模式

（影印版）

Java™ Design Patterns

A Tutorial

（美）James W. Cooper 编著

Pearson Education

科学出版社

www.sciencep.com

Java 程序员书库

Java 设计模式

（影印版）

Java^{TM} Design Patterns

A Tutorial

（美） James W. Cooper 编著

科学出版社

北 京

图字：01-2003-7656 号

内 容 简 介

设计模式已经成为面向对象设计和编程的主要内容，它为解决人们经常遇到的编程问题提供了易于重用且可维护的高级方案。本书介绍了 23 种设计模式，对于每一种模式，都至少提供了一个完整的可视化 Java 程序，使 Java 程序员迅速上手。

本书通俗易懂，方便读者理解设计模式的本质和目的。本书适用于 Java 程序员和自学者。

English reprint copyright©2003 by Science Press and Pearson Education Asia Limited.

Original English language title: Java^{TM} Design Patterns: A Tutorial,1st Edition by James W. Cooper, Copyright©2000
ISBN 0-201-48539-7
All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley Longman, Inc.

For sale and distribution in the People's Republic of China exclusively (except Taiwan, Hong Kong SAR and Macao SAR).
仅限于中华人民共和国境内（不包括中国香港、澳门特别行政区和中国台湾地区）销售发行。

本书封面贴有 Pearson Education(培生教育出版集团)激光防伪标签。无标签者不得销售。

图书在版编目(CIP)数据

Java 设计模式=Java^{TM} Design patterns：A Tutorial （美） James W. Cooper 编著.影印本.—北京：科学出版社，2004
（Java 程序员书库）
ISBN 7-03-012494-4

Ⅰ.J… Ⅱ.C… Ⅲ.Java 语言—程序设计—英文 Ⅳ.TP312

中国版本图书馆 CIP 数据核字（2003）第 103027 号

策划编辑：李佩乾/责任编辑：舒 立
责任印制：吕春珉/封面制作：北新华文平面设计室

科学出版社 出版
北京东黄城根北街16号
邮政编码:100717
http://www.sciencep.com
双青印刷厂 印刷
科学出版社发行 各地新华书店经销
*
2004 年 1 月第一版 开本：787×960 1/16
2004 年 1 月第一次印刷 印张：21 3/4
印数：1—3 000 字数：413 000

定价：37.00 元

（如有印装质量问题，我社负责调换<环伟>）

影印前言

程序设计在于处理复杂性：问题的复杂性和所用的程序设计工具的复杂性。Java 的魅力在于其本身的低复杂性，同时又能很好地处理高度复杂的问题。Java 程序的开发周期只有类似的 C++程序的一半甚至更少，而且 Java 可以方便地处理复杂软件问题：多线程、分布式、跨平台、安全性等。Java 从诞生到现在，已经广泛应用于几乎所有类型软件系统的构建。尤其在基于 Web 的系统开发中，Java 技术具有独特的优势。熟悉 Java 历史的人都知道，Java 的前身——编程语言 Oak 就是致力于电子产品互连的语言，Internet 的发展导致了 Oak 的重生和 Java 的广为流行。现在，J2EE 技术已经成为企业级 Web 应用系统的标准平台。

好的程序设计人员不仅仅要掌握优秀的编程工具，更需要掌握优秀的编程思想。随着面向对象编程技术数十年的发展，人们开始提炼和总结面向对象编程中行之有效的、具有一定普遍意义的方法，即面向对象的设计模式。以 Gamma、Helm、Johnson 和 Vlissides 合著的经典书籍《设计模式》为开端，面向对象设计模式的研究和应用成为面向对象程序设计的重要内容。所有结构良好的面向对象软件系统都包含了大量的设计模式，能否熟练应用设计模式已经成为衡量程序员水平的至关重要的标准。

本丛书收录了与 Java 程序设计和 Java 设计模式相关的经典书籍，反映了应用 Java 开发软件系统的最佳经验总结和最新动态。

《Java 设计模式》采用方便而简洁的编写风格，以可视化的 Java 程序为例，详细介绍了 Gamma、Helm、Johnson 和 Vlissides 合著的经典书籍《设计模式》中列出的所有 23 种模式。通过本书，Java 程序员可以迅速了解和掌握设计模式的内容，并在实践中应用设计模式来创建复杂而健壮的 Java 程序。

《Java 模式应用》介绍了基于模式的开发技巧，演示了使用 Java 开发各种商务系统中的模式应用。书中首先概述设计模式，然后就四种主要模式——创建模式、行为模式、结构模式和系统模式展开了详细的论述。该书还针对系统构建过程中常用的 J2EE、JSP、EJB 和 API 等技术作了介绍，适合具有一定编程基础的 Java 程序员阅读参考。

《J2EE 核心模式》是 Sun Java Center 的资深设计师的 J2EE 亲身实践经验的总结。该书主要描述 J2EE 关键技术（Java Server Pages，Java Servlet，Enterprise Java Beans，Java Message Services 等）的模式、最佳实践、设计策略和经过验证的解决方案。该书介绍了 J2EE 包括的 15 个模式的分类和大量的策略，不仅具有理论深度，而且非常实用。该书内容适合 J2EE 的爱好者、程序员、设计师、开发者和技术管理者。一句话，该书

适合于设计、构建和开发 J2EE 平台应用的所有人。

XML 是新一代文档的标准，Web 页、数据、源码等等，均可以用 XML 文档表示。越来越多的程序员正在使用 Java 来处理 XML 文档。《用 Java 处理 XML》详细论述了如何使用 Java 来读写 XML 文档。该书是目前最新和最全的 Java 处理 XML 技术的介绍，包含内容超过 1000 页的关于 SAX, DOM, JDOM, JAXP, TrAX, XPath, XSLT, SOAP 等的讲解。该书适合于使用 Java 读写 XML 文档的 Java 程序员。其内容从基本概念到高级应用无所不包，特别适合作为手册随时参考。

在《实时 Java》中，作为 RTSJ 专家组的成员之一，Dibble 从 Java 平台特有的实时问题概述开始，依次讲解了 RTSJ 各项主要特性的使用方法。从广泛的实时原理到详细的编程隐患，该书覆盖了构建有效实时程序所需的一切知识。其主要内容包括：与非实时代码的互操作性、实时开发中的取舍以及 JVM 软件的实时问题；垃圾收集、无堆栈访问、物理内存和“不朽”内存以及无堆栈内存的常数时间分配；优先级调度、期限调度以及速率单调分析；闭包、异步传输控制、异步事件以及计时器。这是一本非常实用的指南，适用于有经验的 Java 平台开发人员。

《Java 数据结构与算法分析》介绍了常见的数据结构，如链表、堆栈、队列、树和哈希表等，并对查找和排序进行了算法分析，给出了相应的 Java 实现。该书逻辑结构严谨，主次分明，可用作程序员参考书。

总之，这套书详细介绍了 Java 应用的许多重要方面：从具有普遍性的 Java 数据结构和算法、Java 设计模式、Java 模式应用、J2EE 核心模式，到日益显著的 Java 特殊应用领域（Java 处理 XML 文档和 Java 实时系统开发）。其内容具有一定的理论深度，更有重要的实际参考价值。

有鉴于此，特向 Java 系统开发和应用领域中不同程度的读者推荐这套书，相信每位有心的读者都能得到物超所值的收获。

清华大学经济管理学院管理科学与工程系　朱涛　博士
讲授课程：Java 程序设计，面向对象的分析设计方法

敬告读者：

本书光盘如有需求，请向我社索取，费用 20.00 元/张。

联系人：朱敏　电话：010-62622941　E-mail: zhumin@abook.cn

试读结束：需要全本请在线购买： www.ertongbook.com

PREFACE

This is a practical book that tells you how to write Java programs using some of the most common *design patterns.* It is structured as a series of short chapters, each describing a design pattern and giving one or more complete, working, visual example programs that use that pattern. Each chapter also includes Unified Modeling Language (UML) diagrams illustrating how the classes interact.

This book is not a "companion" book to the well-known *Design Patterns* text [Gamma, 1995] by the "Gang of Four." Rather, it is a tutorial for people who want to learn what design patterns are about and how to use them in their work. You need not have read *Design Patterns* to gain from reading this book, but when you are done here you might want to read or reread that book to gain additional insights.

In this book, you will learn that design patterns are a common way to organize objects in your programs to make those programs easier to write and modify. You'll also see that by familiarizing yourself with these design patterns, you will gain a valuable vocabulary for discussing how your programs are constructed.

People come to appreciate design patterns in different ways—from the highly theoretical to the intensely practical—and when they finally see the great power of these patterns, they experience an "Aha!" moment. Usually this moment means that you suddenly had an internal picture of how that pattern can help you in your work.

In this book, we try to help you form that conceptual idea, or *gestalt,* by describing the pattern in as many ways as possible. The book is organized into six main sections:

- An introductory description
- A description of patterns grouped into three sections: Creational, Structural, and Behavioral

- A description of the Java Foundation Classes (JFC) showing the patterns they illustrate
- A set of case studies where patterns have been helpful

For each pattern, we start with a brief verbal description and then build simple example programs. Each example is a visual program that you can run and examine so as to make the pattern as concrete as possible. All of the example programs and their variations are on the CD-ROM that accompanies this book. In that way, you can run them, change them, and see how the variations that you create work.

All of the programs are based on Java 1.2, and most use the JFC. If you haven't taken the time to learn how to use these classes, there is a tutorial covering the basics in Appendix A where we also discuss some of the patterns that they illustrate.

Since each of the examples consists of a number of Java files for each of the classes we use in that example, we also provide a Visual SlickEdit project file for each example and place each example in a separate subdirectory to prevent any confusion.

As you leaf through the book, you'll see screen shots of the programs we developed to illustrate the design patterns; these provide yet another way to reinforce your learning of these patterns. You'll also see UML diagrams of these programs that illustrate the interactions between classes in yet another way. UML diagrams are just simple box and arrow illustrations of classes and their inheritance structure, with the arrows pointing to parent classes and dotted arrows pointing to interfaces. If you are unfamiliar with UML, we provide a simple introduction in the first chapter.

Finally, since we used JVISION to create the UML diagrams in each chapter, we provide the original JVISION diagram files for each pattern as well, so you can use your own copy of JVISION to play with them.

When you finish this book, you'll be comfortable with the basics of design patterns and will be able to start using them in your day to day Java programming work.

James W. Cooper
Wilton, CT
Nantucket, MA
November, 1999

ACKNOWLEDGMENTS

Writing a book on Java design patterns has been a fascinating challenge. Design patterns are intellectually recursive; that is, every time that you think that you've wrapped up a good explanation of one, another turn of the crank occurs to you or is suggested by someone else. So, while writing is essentially a solitary task, I had a lot of help and support.

Foremost, I thank Roy Byrd and Alan Marwick of IBM Research for encouraging me to tackle this book and providing lots of support during the manuscript's genesis and revision. I also especially thank Nicole Cooper for editing my first draft; she definitely improved its clarity and accuracy.

The design pattern community (informally called the "Pattern-nostra") were also a great help. In particular, I thank both John Vlissides and Ken Arnold for their careful and thoughtful reading of the manuscript. Among the many others, I thank Ralph Johnson, Sherman Alpert, Zunaid Kazi, Colin Harrison, and Hank Stuck. I'm also grateful to John Dorsey and Tyler Sperry at *JavaPro* magazine for their encouragement and editorial suggestions on some of the columns that I wrote that later became parts of this book. Thanks also to Herb Chong and Mary Neff for lending their names and part of their project descriptions to the case studies chapter. Finally, thanks to my wife Vicki, who provided endless support during the ups and downs of endless writing and seemingly endless revision.

CONTENTS

Preface *xiii*
Acknowledgments *xv*

Section 1 What Are Design Patterns? 1

Chapter 1 Introduction 3
Defining Design Patterns 5
The Learning Process 6
Studying Design Patterns 7
Notes on Object-Oriented Approaches 7
The Java Foundation Classes 8
Java Design Patterns 9

Chapter 2 UML Diagrams 11
Inheritance 12
Interfaces 13
Composition 14
Annotation 15
JVISION UML Diagrams 15
Visual SlickEdit Project Files 15

Section 2 Creational Patterns 17

Chapter 3 The Factory Pattern 19
How a Factory Works 19
Sample Code 20

The Two Subclasses 20
Building the Simple Factory 21
Factory Patterns in Math Computation 23
Thought Questions 24
Programs on the CD-ROM 24

Chapter 4 The Factory Method **25**
The Swimmer Class 27
The Event Classes 27
Straight Seeding 28
Our Seeding Program 30
Other Factories 30
When to Use a Factory Method 31
Thought Question 31
Programs on the CD-ROM 31

Chapter 5 The Abstract Factory Pattern **33**
A GardenMaker Factory 34
How the User Interface Works 36
Adding More Classes 37
Consequences of the Abstract Factory Pattern 37
Thought Question 38
Programs on the CD-ROM 38

Chapter 6 The Singleton Pattern **39**
Creating a Singleton Using a Static Method 39
Exceptions and Instances 40
Throwing an Exception 41
Creating an Instance of the Class 41
Providing a Global Point of Access to a Singleton Pattern 42
The javax.comm Package as a Singleton 43
Other Consequences of the Singleton Pattern 46
Thought Question 47
Programs on the CD-ROM 47

Chapter 7 The Builder Pattern	49
An Investment Tracker	50
Calling the Builders	52
The List Box Builder	54
The Check Box Builder	54
Consequences of the Builder Pattern	55
Thought Questions	56
Programs on the CD-ROM	56
Chapter 8 The Prototype Pattern	57
Cloning in Java	58
Using the Prototype	58
Using the Prototype Pattern	61
Prototype Managers	64
Cloning Using Serialization	65
Consequences of the Prototype Pattern	66
Thought Question	66
Programs on the CD-ROM	67
Summary of Creational Patterns	67
Section 3 Structural Patterns	**69**
Chapter 9 The Adapter Pattern	71
Moving Data between Lists	71
Using the JFC JList Class	73
Two-Way Adapters	78
Pluggable Adapters	78
Adapters in Java	79
Thought Question	81
Programs on the CD-ROM	81
Chapter 10 The Bridge Pattern	83
The Class Diagram	85.
Extending the Bridge	86

试读结束：需要全本请在线购买： www.ertongbook.com

Java Beans as Bridges 87
Consequences of the Bridge Pattern 88
Thought Question 89
Programs on the CD-ROM 89

Chapter 11 The Composite Pattern **91**
An Implementation of a Composite 92
Computing Salaries 93
The Employee Classes 93
The Boss Class 95
Building the Employee Tree 97
Self-Promotion 98
Doubly Linked List 99
Consequences of the Composite Pattern 100
A Simple Composite 100
Composites in Java 100
Other Implementation Issues 101
Thought Questions 101
Programs on the CD-ROM 101

Chapter 12 The Decorator Pattern **103**
Decorating a CoolButton 103
Using a Decorator 105
The Class Diagram 107
Decorating Borders in Java 108
Nonvisual Decorators 109
Decorators, Adapters, and Composites 112
Consequences of the Decorator Pattern 113
Thought Questions 113
Programs on the CD-ROM 113

Chapter 13 The Façade Pattern **115**
Building the Façade Classes 117
Consequences of the Façade Pattern 120
Notes on Installing and Running the dbFrame Program 121

Thought Question 122
Programs on the CD-ROM 122

Chapter 14 The Flyweight Pattern 123
Discussion 124
Example Code 124
Flyweight Uses in Java 129
Sharable Objects 130
Copy-on-Write Objects 130
Thought Question 131
Programs on the CD-ROM 131

Chapter 15 The Proxy Pattern 133
Sample Code 134
Copy-on-Write 136
Enterprise Java Beans 136
Comparison with Related Patterns 137
Thought Question 137
Programs on the CD-ROM 137
Summary of Structural Patterns 137

Section 4 Behavioral Patterns 139

Chapter 16 Chain of Responsibility Pattern 141
Applicability 142
Sample Code 143
The List Boxes 145
Programming a Help System 148
A Chain or a Tree? 151
Kinds of Requests 152
Examples in Java 152
Consequences of the Chain of Responsibility 153
Thought Questions 153
Programs on the CD-ROM 154

Chapter 17 The Command Pattern 155
Motivation 155
Command Objects 156
Building Command Objects 157
The Command Pattern 159
The Command Pattern in the Java Language 161
Consequences of the Command Pattern 162
Providing Undo 163
Thought Questions 167
Programs on the CD-ROM 168

Chapter 18 The Interpreter Pattern 169
Motivation 169
Applicability 170
Simple Report Example 170
Interpreting the Language 171
Objects Used in Parsing 172
Reducing the Parsed Stack 175
Implementing the Interpreter Pattern 176
Consequences of the Interpreter Pattern 180
Thought Question 181
Programs on the CD-ROM 181

Chapter 19 The Iterator Pattern 183
Motivation 183
Enumerations in Java 184
Sample Code 184
Filtered Iterators 185
Consequence of the Iterator Pattern 188
Composites and Iterators 189
Iterators in Java 1.2 189
Thought Question 189
Programs on the CD-ROM 190

Chapter 20 The Mediator Pattern **191**
An Example System 191
Interactions between Controls 193
Sample Code 194
Mediators and Command Objects 197
Consequences of the Mediator Pattern 198
Single Interface Mediators 198
Implementation Issues 199
Programs on the CD-ROM 199

Chapter 21 The Memento Pattern **201**
Motivation 201
Implementation 202
Sample Code 203
Consequences of the Memento Pattern 207
Thought Question 208
Programs on the CD-ROM 208

Chapter 22 The Observer Pattern **209**
Watching Colors Change 210
The Message to the Media 213
The JList as an Observer 214
The MVC Architecture as an Observer 216
The Observer Interface and Observable Class 216
Consequences of the Observer Pattern 216
Thought Questions 217
Programs on the CD-ROM 217

Chapter 23 The State Pattern **219**
Sample Code 219
Switching between States 224
How the Mediator Interacts with the StateManager 225
State Transitions 227
Mediators and the God Class 227

Consequences of the State Pattern 227
Thought Questions 228
Programs on the CD-ROM 228

Chapter 24 The Strategy Pattern **229**
Motivation 229
Sample Code 230
The Context Class 231
The Program Commands 232
The Line and Bar Graph Strategies 233
Drawing Plots in Java 233
Consequences of the Strategy Pattern 236
Thought Question 237
Programs on the CD-ROM 237

Chapter 25 The Template Pattern **239**
Motivation 239
Kinds of Methods in a Template Class 241
Template Method Patterns in Java 241
Sample Code 242
Templates and Callbacks 246
Consequences of the Template Pattern 247
Thought Question 247
Programs on the CD-ROM 247

Chapter 26 The Visitor Pattern **249**
Motivation 249
When to Use the Visitor Pattern 251
Sample Code 251
Visiting the Classes 253
Visiting Several Classes 253
Bosses are Employees, Too 255
Catch-All Operations Using Visitors 256
Double Dispatching 257
Traversing a Series of Classes 257

Consequence of the Visitor Pattern 257
Thought Question 258
Programs on the CD-ROM 258

Section 5 Design Patterns and the Java Foundation Classes 259

Chapter 27 The JFC, or Swing **261**
Installing and Using Swing 261
Ideas behind Swing 262
The Swing Class Hierarchy 262

Chapter 28 Writing a Simple JFC Program **263**
Setting the Look and Feel 263
Setting the Window Close Box 264
Making a JxFrame Class 264
A Simple Two-Button Program 265
More on JButton 266
Programs on the CD-ROM 267

Chapter 29 Radio Buttons and Toolbars **269**
Radio Buttons 269
The JToolBar 270
JToggleButton 270
A Sample Button Program 271
Programs on the CD-ROM 271

Chapter 30 Menus and Actions **273**
Action Objects 273
Design Patterns in the Action Object 276
Programs on the CD-ROM 277

Chapter 31 The JList Class **279**
List Selections and Events 280
Changing a List Display Dynamically 281
A Sorted JList with a ListModel 282

试读结束：需要全本请在线购买： www.ertongbook.com