

高等学校设计模式课程系列教材

内容丰富，详细讲解7个面向对象设计原则和24个常用设计模式实例驱动，提供24个模式应用实例的完整设计方案和程序代码学练结合，包含超过150道习题和3套设计模式模拟试题

# C# 设计模式

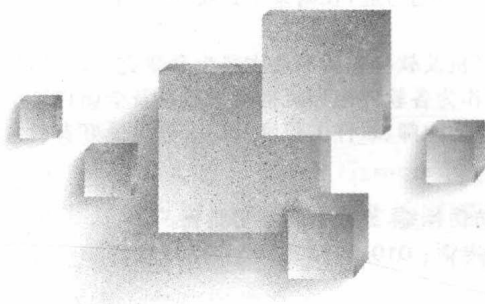
## (第2版)

◎ 刘伟 胡志刚 编著

清华大学出版社



高等学校设计模式课程系列教材



# C# 设计模式

## (第2版)

---

◎ 刘伟 胡志刚 编著

清华大学出版社  
北京

## 内 容 简 介

本书使用 C# 语言详细讲解了 24 种设计模式。全书共分为 26 章,内容包括设计模式概述、面向对象设计原则、简单工厂模式、工厂方法模式、抽象工厂模式、建造者模式、原型模式、单例模式、适配器模式、桥接模式、组合模式、装饰模式、外观模式、享元模式、代理模式、职责链模式、命令模式、解释器模式、迭代器模式、中介者模式、备忘录模式、观察者模式、状态模式、策略模式、模板方法模式和访问者模式。

本书结合大量项目应用实例,对每一个设计模式都进行了细致的讲解。本书结构合理、条理清晰、内容丰富,且在每一章后面都配有大量的习题,在附录中还提供了对 UML 类图相关知识的介绍和三套设计模式模拟试题。

本书既可作为高等院校计算机及软件相关专业本科生和研究生软件设计模式、软件体系结构、面向对象分析与设计等课程教材,也可作为各软件培训机构培训教材及全国计算机技术与软件专业技术资格(水平)考试辅导教材,还可作为软件架构师、软件工程师及广大软件爱好者的自学读本和参考用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

C# 设计模式/刘伟,胡志刚编著.—2 版.—北京:清华大学出版社,2018(2018.8重印)

(高等学校设计模式课程系列教材)

ISBN 978-7-302-48570-4

I. ①C… II. ①刘… ②胡… III. ①C 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 241133 号

责任编辑:魏江江

封面设计:刘 键

责任校对:白 蕾

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京泽宇印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:27.25 字 数:662 千字

版 次:2013 年 5 月第 1 版 2018 年 1 月第 2 版 印 次:2018 年 8 月第 2 次印刷

印 数:8501~9500

定 价:69.50 元

产品编号:072617-01

设计模式和敏捷开发方法的创始人之一 Erich Gamma 曾说过：“设计和开发面向对象软件是非常困难的，而设计和开发可复用的面向对象软件则更加困难”。正因为如此，在软件开发过程中，有经验的设计者往往会重复使用一些他们在以前的设计工作中曾经用到的成功而有效的解决方案，这些解决方案可以提高开发人员的开发效率与软件质量，并使所设计的软件更加灵活、易于扩展，可复用性也更高。这些解决方案即为设计模式，设计模式为实现可维护性复用而诞生，无论是面向对象编程的初学者还是有一定编程经验的程序员，都可以从设计模式的学习和使用中深入了解面向对象思想的精华，开发出可扩展性和可复用性俱佳的软件。

设计模式是前人经验的积累，它让我们的软件变得更像一个艺术品，而不是一堆难以维护和重用的代码，设计模式已经成功地应用于众多软件设计中，并且成为一名优秀的面向对象软件开发人员所必须掌握的知识和技能。本书编者在十多年的软件开发和计算机教育教学中积累了丰富的设计模式使用经验和教学经验，也深刻体会到学习设计模式的重要性。目前，国内外越来越多的高校在计算机及软件工程相关专业的本科生和研究生教学中开设了软件体系结构、面向对象分析与设计等课程，而设计模式是这些课程的核心组成部分；有的学校还将设计模式作为一门单独的课程来开设。此外，很多软件培训机构在软件工程师培训课程中也包含了设计模式的相关内容，在各类计算机考试和企业招聘笔试中，设计模式相关考题也占据了一定的比例。为此，编写一本适合教学所需的设计模式教材是一件非常有意义的事情。编者在成功完成《设计模式》《设计模式实训教程》等书籍的基础上，针对目前部分高校对 C# 设计模式教材的需求，特编写了本书。

本书的编写目的在于让读者通过大量的应用实例和习题，理论联系实际，以便更快、更好地理解 and 掌握每一个设计模式。本书共 26 章，可分为 4 个部分：

第 1 部分包含第 1 章和第 2 章，主要介绍与设计模式相关的一些基础知识，包括设计模式概述、7 个常用的面向对象设计原则等内容，为后续设计模式的学习奠定基础。

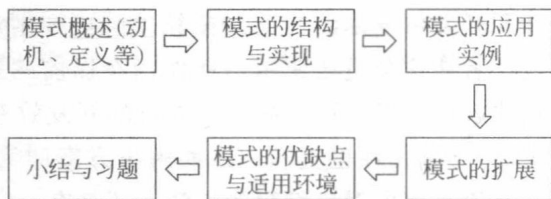
第 2 部分包含第 3~8 章，介绍 6 种常用的创建型设计模式，分别是简单工厂模式、工厂方法模式、抽象工厂模式、建造者模式、原型模式和单例模式。

第 3 部分包含第 9~15 章，介绍 7 种常用的结构型设计模式，分别是适配器模式、桥接模式、组合模式、装饰模式、外观模式、享元模式和代理模式。

第 4 部分包含第 16~26 章，介绍 11 种常用的行为型设计模式，分别是职责链模式、命令模式、解释器模式、迭代器模式、中介者模式、备忘录模式、观察者模式、状态模式、策略模式、模板方法模式和访问者模式。

本书结合大量的项目应用实例,对每一个设计模式都进行了细致的讲解。本书结构合理、条理清晰、内容丰富、讲解深入,且在每一章后面都配有大量的习题,很多习题都基于某个项目实例,让读者在学完相关知识后能够更好地消化和巩固所学知识。此外,在附录中还提供了对 UML 类图相关知识的介绍和三套设计模式模拟试题。

本书在编写过程中参考了大量已有的设计模式书籍,集各家之所长,并进行了适当的整理和扩展,将一些原本深奥并难以理解的设计思想通过一些简单的应用实例来解析,让读者能够轻松掌握面向对象设计思想的精髓。本书以“实例驱动教学”为整体编写原则,对于每一个模式的学习都至少基于一个应用实例,结合应用实例来分析和讲解每一个设计模式,力求通过最通俗易懂的方式让读者学习和理解设计模式,让读者在真实项目实例的引导下学会选择和合理运用设计模式。在设计模式讲解部分,每一章的基本结构如下:



本书提供了完整的设计模式及应用实例的 UML 结构图(类图)和 C# 实现代码,所有类图均严格按照 UML 2. X 标准绘制,所有代码均在 Microsoft Visual Studio 2010 中通过测试且运行无误。本书提供了完整的配套教学资料,包括所有实例的源代码、PPT 格式的电子课件、课后习题和模拟试题答案等,这些教学资料形成一个完整的体系,为教学和学习提供便利。

本书既可作为高等院校计算机及软件相关专业本科生和研究生软件设计模式、软件体系结构、面向对象分析与设计等课程教材,也可作为各软件培训机构培训教材及全国计算机技术与软件专业技术资格(水平)考试辅导教材,还可作为软件架构师、软件工程师及广大软件爱好者的自学读本和参考用书。

本书在第一版的基础上进行了修订和完善,修改了第一版中存在的一些问题,并在附录中增加了一套全新的模拟试题。在此,向所有对本书第一版提出宝贵意见和建议的同仁们表示感谢。

由于时间仓促、编者学识有限,虽经多次审阅与校稿,但书中仍可能存在不足和疏漏之处,恳请广大读者将意见和建议通过清华大学出版社反馈给我们。

编者

2017年10月

<b>第 1 章 设计模式概述</b> .....	1
1.1 设计模式的诞生与发展 .....	1
1.1.1 模式的诞生与定义 .....	1
1.1.2 软件模式概述 .....	3
1.1.3 设计模式的发展 .....	3
1.2 设计模式的定义与分类 .....	4
1.2.1 设计模式的定义 .....	4
1.2.2 设计模式的基本要素 .....	5
1.2.3 设计模式的分类 .....	6
1.3 GoF 设计模式简介 .....	7
1.4 设计模式的优点 .....	9
1.5 本章小结 .....	10
1.6 习题 .....	10
<b>第 2 章 面向对象设计原则</b> .....	12
2.1 面向对象设计原则概述 .....	12
2.2 单一职责原则 .....	13
2.3 开闭原则 .....	15
2.4 里氏代换原则 .....	15
2.5 依赖倒转原则 .....	16
2.6 接口隔离原则 .....	18
2.7 合成复用原则 .....	20
2.8 迪米特法则 .....	22
2.9 本章小结 .....	24
2.10 习题 .....	24
<b>第 3 章 简单工厂模式</b> .....	27
3.1 创建型模式 .....	27
3.2 简单工厂模式概述 .....	28

3.3	简单工厂模式的结构与实现	29
3.3.1	简单工厂模式的结构	29
3.3.2	简单工厂模式的实现	30
3.4	简单工厂模式的应用实例	31
3.5	创建对象与使用对象	36
3.6	简单工厂模式的简化	39
3.7	简单工厂模式的优缺点与适用环境	39
3.7.1	简单工厂模式的优点	39
3.7.2	简单工厂模式的缺点	39
3.7.3	简单工厂模式的适用环境	40
3.8	本章小结	40
3.9	习题	41
<b>第4章</b>	<b>工厂方法模式</b>	<b>43</b>
4.1	工厂方法模式概述	43
4.2	工厂方法模式的结构与实现	45
4.2.1	工厂方法模式的结构	45
4.2.2	工厂方法模式的实现	45
4.3	工厂方法模式的应用实例	46
4.4	配置文件与反射	50
4.5	工厂方法的重载	51
4.6	工厂方法的隐藏	53
4.7	工厂方法模式的优缺点与适用环境	54
4.7.1	工厂方法模式的优点	54
4.7.2	工厂方法模式的缺点	55
4.7.3	工厂方法模式的适用环境	55
4.8	本章小结	55
4.9	习题	56
<b>第5章</b>	<b>抽象工厂模式</b>	<b>58</b>
5.1	产品等级结构与产品族	58
5.2	抽象工厂模式概述	59
5.3	抽象工厂模式的结构与实现	60
5.3.1	抽象工厂模式的结构	60
5.3.2	抽象工厂模式的实现	61
5.4	抽象工厂模式的应用实例	62
5.5	开闭原则的倾斜性	69
5.6	抽象工厂模式的优缺点与适用环境	69
5.6.1	抽象工厂模式的优点	69

5.6.2	抽象工厂模式的缺点	70
5.6.3	抽象工厂模式的适用环境	70
5.7	本章小结	70
5.8	习题	71
<b>第6章</b>	<b>建造者模式</b>	<b>73</b>
6.1	建造者模式概述	73
6.2	建造者模式的结构与实现	74
6.2.1	建造者模式的结构	74
6.2.2	建造者模式的实现	75
6.3	建造者模式的应用实例	78
6.4	指挥者类的深入讨论	84
6.5	建造者模式的优缺点与适用环境	88
6.5.1	建造者模式的优点	88
6.5.2	建造者模式的缺点	88
6.5.3	建造者模式的适用环境	88
6.6	本章小结	89
6.7	习题	89
<b>第7章</b>	<b>原型模式</b>	<b>91</b>
7.1	原型模式概述	91
7.2	原型模式的结构与实现	92
7.2.1	原型模式的结构	92
7.2.2	浅克隆与深克隆	93
7.2.3	原型模式的实现	94
7.3	原型模式的应用实例	97
7.4	原型管理器	101
7.5	原型模式的优缺点与适用环境	102
7.5.1	原型模式的优点	102
7.5.2	原型模式的缺点	102
7.5.3	原型模式的适用环境	102
7.6	本章小结	103
7.7	习题	103
<b>第8章</b>	<b>单例模式</b>	<b>105</b>
8.1	单例模式概述	105
8.2	单例模式的结构与实现	106
8.2.1	单例模式的结构	106
8.2.2	单例模式的实现	107



8.3	单例模式的应用实例 .....	108
8.4	饿汉式单例与懒汉式单例 .....	111
8.5	单例模式的优缺点与适用环境 .....	114
8.5.1	单例模式的优点 .....	114
8.5.2	单例模式的缺点 .....	114
8.5.3	单例模式的适用环境 .....	114
8.6	本章小结 .....	115
8.7	习题 .....	115
<b>第9章</b>	<b>适配器模式 .....</b>	<b>117</b>
9.1	结构型模式 .....	117
9.2	适配器模式概述 .....	118
9.3	适配器模式的结构与实现 .....	119
9.3.1	适配器模式的结构 .....	119
9.3.2	适配器模式的实现 .....	120
9.4	适配器模式的应用实例 .....	122
9.5	缺省适配器模式 .....	127
9.6	双向适配器 .....	127
9.7	适配器模式的优缺点与适用环境 .....	128
9.7.1	适配器模式的优点 .....	129
9.7.2	适配器模式的缺点 .....	129
9.7.3	适配器模式的适用环境 .....	129
9.8	本章小结 .....	130
9.9	习题 .....	130
<b>第10章</b>	<b>桥接模式 .....</b>	<b>132</b>
10.1	桥接模式概述 .....	132
10.2	桥接模式的结构与实现 .....	134
10.2.1	桥接模式的结构 .....	134
10.2.2	桥接模式的实现 .....	134
10.3	桥接模式的应用实例 .....	136
10.4	桥接模式与适配器模式联用 .....	142
10.5	桥接模式的优缺点与适用环境 .....	143
10.5.1	桥接模式的优点 .....	143
10.5.2	桥接模式的缺点 .....	143
10.5.3	桥接模式的适用环境 .....	144
10.6	本章小结 .....	144
10.7	习题 .....	144

第 11 章 组合模式 .....	146
11.1 组合模式概述 .....	146
11.2 组合模式的结构与实现 .....	148
11.2.1 组合模式的结构 .....	148
11.2.2 组合模式的实现 .....	148
11.3 组合模式的应用实例 .....	150
11.4 透明组合模式与安全组合模式 .....	156
11.5 组合模式的优缺点与适用环境 .....	157
11.5.1 组合模式的优点 .....	158
11.5.2 组合模式的缺点 .....	158
11.5.3 组合模式的适用环境 .....	158
11.6 本章小结 .....	158
11.7 习题 .....	159
第 12 章 装饰模式 .....	161
12.1 装饰模式概述 .....	161
12.2 装饰模式的结构与实现 .....	162
12.2.1 装饰模式的结构 .....	162
12.2.2 装饰模式的实现 .....	163
12.3 装饰模式的应用实例 .....	165
12.4 透明装饰模式与半透明装饰模式 .....	170
12.5 装饰模式的优缺点与适用环境 .....	171
12.5.1 装饰模式的优点 .....	171
12.5.2 装饰模式的缺点 .....	172
12.5.3 装饰模式的适用环境 .....	172
12.6 本章小结 .....	172
12.7 习题 .....	173
第 13 章 外观模式 .....	175
13.1 外观模式概述 .....	175
13.2 外观模式的结构与实现 .....	177
13.2.1 外观模式的结构 .....	177
13.2.2 外观模式的实现 .....	177
13.3 外观模式的应用实例 .....	179
13.4 抽象外观类 .....	183
13.5 外观模式的优缺点与适用环境 .....	186
13.5.1 外观模式的优点 .....	186
13.5.2 外观模式的缺点 .....	187

13.5.3 外观模式的适用环境 .....	187
13.6 本章小结 .....	187
13.7 习题 .....	188
<b>第 14 章 享元模式 .....</b>	<b>190</b>
14.1 享元模式概述 .....	190
14.2 享元模式的结构与实现 .....	191
14.2.1 享元模式的结构 .....	191
14.2.2 享元模式的实现 .....	192
14.3 享元模式的应用实例 .....	194
14.4 有外部状态的享元模式 .....	198
14.5 单纯享元模式和复合享元模式 .....	201
14.6 享元模式的优缺点与适用环境 .....	202
14.6.1 享元模式的优点 .....	202
14.6.2 享元模式的缺点 .....	202
14.6.3 享元模式的适用环境 .....	202
14.7 本章小结 .....	202
14.8 习题 .....	203
<b>第 15 章 代理模式 .....</b>	<b>205</b>
15.1 代理模式概述 .....	205
15.2 代理模式的结构与实现 .....	206
15.2.1 代理模式的结构 .....	206
15.2.2 代理模式的实现 .....	207
15.3 代理模式的应用实例 .....	209
15.4 远程代理 .....	213
15.5 虚拟代理 .....	214
15.6 缓冲代理 .....	214
15.7 代理模式的优缺点与适用环境 .....	216
15.7.1 代理模式的优点 .....	216
15.7.2 代理模式的缺点 .....	217
15.7.3 代理模式的适用环境 .....	217
15.8 本章小结 .....	217
15.9 习题 .....	218
<b>第 16 章 职责链模式 .....</b>	<b>220</b>
16.1 行为型模式 .....	221
16.2 职责链模式概述 .....	222
16.3 职责链模式的结构与实现 .....	223

16.3.1	职责链模式的结构 .....	223
16.3.2	职责链模式的实现 .....	224
16.4	职责链模式的应用实例 .....	225
16.5	纯与不纯的职责链模式 .....	232
16.6	职责链模式的优缺点与适用环境 .....	232
16.6.1	职责链模式的优点 .....	232
16.6.2	职责链模式的缺点 .....	233
16.6.3	职责链模式的适用环境 .....	233
16.7	本章小结 .....	233
16.8	习题 .....	234
<b>第 17 章</b>	<b>命令模式 .....</b>	<b>236</b>
17.1	命令模式概述 .....	236
17.2	命令模式的结构与实现 .....	237
17.2.1	命令模式的结构 .....	237
17.2.2	命令模式的实现 .....	238
17.3	命令模式的应用实例 .....	240
17.4	实现命令队列 .....	244
17.5	记录请求日志 .....	245
17.6	实现撤销操作 .....	246
17.7	宏命令 .....	249
17.8	命令模式的优缺点与适用环境 .....	249
17.8.1	命令模式的优点 .....	249
17.8.2	命令模式的缺点 .....	250
17.8.3	命令模式的适用环境 .....	250
17.9	本章小结 .....	251
17.10	习题 .....	251
<b>第 18 章</b>	<b>解释器模式 .....</b>	<b>253</b>
18.1	解释器模式概述 .....	253
18.2	语法规则和抽象语法树 .....	254
18.3	解释器模式的结构与实现 .....	255
18.3.1	解释器模式的结构 .....	255
18.3.2	解释器模式的实现 .....	256
18.4	解释器模式的应用实例 .....	258
18.5	解释器模式的优缺点与适用环境 .....	265
18.5.1	解释器模式的优点 .....	265
18.5.2	解释器模式的缺点 .....	265
18.5.3	解释器模式的适用环境 .....	265

18.6 本章小结 .....	266
18.7 习题 .....	266
<b>第 19 章 迭代器模式 .....</b>	<b>268</b>
19.1 迭代器模式概述 .....	268
19.2 迭代器模式的结构与实现 .....	269
19.2.1 迭代器模式的结构 .....	269
19.2.2 迭代器模式的实现 .....	270
19.3 迭代器模式的应用实例 .....	272
19.4 使用内部类实现迭代器 .....	278
19.5 .NET 内置迭代器 .....	279
19.6 迭代器模式的优缺点与适用环境 .....	282
19.6.1 迭代器模式的优点 .....	282
19.6.2 迭代器模式的缺点 .....	282
19.6.3 迭代器模式的适用环境 .....	283
19.7 本章小结 .....	283
19.8 习题 .....	283
<b>第 20 章 中介者模式 .....</b>	<b>285</b>
20.1 中介者模式概述 .....	285
20.2 中介者模式的结构与实现 .....	287
20.2.1 中介者模式的结构 .....	287
20.2.2 中介者模式的实现 .....	288
20.3 中介者模式的应用实例 .....	290
20.4 扩展中介者与同事类 .....	296
20.5 中介者模式的优缺点与适用环境 .....	300
20.5.1 中介者模式的优点 .....	300
20.5.2 中介者模式的缺点 .....	300
20.5.3 中介者模式的适用环境 .....	300
20.6 本章小结 .....	301
20.7 习题 .....	301
<b>第 21 章 备忘录模式 .....</b>	<b>303</b>
21.1 备忘录模式概述 .....	303
21.2 备忘录模式的结构与实现 .....	304
21.2.1 备忘录模式的结构 .....	304
21.2.2 备忘录模式的实现 .....	305
21.3 备忘录模式的应用实例 .....	308
21.4 实现多次撤销 .....	312

21.5	备忘录模式的优缺点与适用环境 .....	315
21.5.1	备忘录模式的优点 .....	315
21.5.2	备忘录模式的缺点 .....	315
21.5.3	备忘录模式的适用环境 .....	315
21.6	本章小结 .....	316
21.7	习题 .....	316
<b>第 22 章</b>	<b>观察者模式 .....</b>	<b>318</b>
22.1	观察者模式概述 .....	318
22.2	观察者模式的结构与实现 .....	319
22.2.1	观察者模式的结构 .....	319
22.2.2	观察者模式的实现 .....	320
22.3	观察者模式的应用实例 .....	322
22.4	观察者模式与 .NET 中的委托事件模型 .....	327
22.5	观察者模式与 MVC .....	330
22.6	观察者模式的优缺点与适用环境 .....	330
22.6.1	观察者模式的优点 .....	330
22.6.2	观察者模式的缺点 .....	331
22.6.3	观察者模式的适用环境 .....	331
22.7	本章小结 .....	331
22.8	习题 .....	332
<b>第 23 章</b>	<b>状态模式 .....</b>	<b>333</b>
23.1	状态模式概述 .....	333
23.2	状态模式的结构与实现 .....	334
23.2.1	状态模式的结构 .....	334
23.2.2	状态模式的实现 .....	335
23.3	状态模式的应用实例 .....	337
23.4	共享状态 .....	344
23.5	使用环境类实现状态的转换 .....	348
23.6	状态模式的优缺点与适用环境 .....	351
23.6.1	状态模式的优点 .....	351
23.6.2	状态模式的缺点 .....	351
23.6.3	状态模式的适用环境 .....	352
23.7	本章小结 .....	352
23.8	习题 .....	352
<b>第 24 章</b>	<b>策略模式 .....</b>	<b>355</b>
24.1	策略模式概述 .....	355

24.2	策略模式的结构与实现 .....	356
24.2.1	策略模式的结构 .....	356
24.2.2	策略模式的实现 .....	357
24.3	策略模式的应用实例 .....	358
24.4	PetShop 4.0 中的订单处理 .....	362
24.5	策略模式的优缺点与适用环境 .....	363
24.5.1	策略模式的优点 .....	363
24.5.2	策略模式的缺点 .....	364
24.5.3	策略模式的适用环境 .....	364
24.6	本章小结 .....	364
24.7	习题 .....	365
<b>第 25 章</b>	<b>模板方法模式 .....</b>	<b>366</b>
25.1	模板方法模式概述 .....	366
25.2	模板方法模式的结构与实现 .....	367
25.2.1	模板方法模式的结构 .....	367
25.2.2	模板方法模式的实现 .....	368
25.3	模板方法模式的应用实例 .....	371
25.4	钩子方法的使用 .....	374
25.5	模板方法模式的优缺点与适用环境 .....	377
25.5.1	模板方法模式的优点 .....	377
25.5.2	模板方法模式的缺点 .....	378
25.5.3	模板方法模式的适用环境 .....	378
25.6	本章小结 .....	378
25.7	习题 .....	379
<b>第 26 章</b>	<b>访问者模式 .....</b>	<b>380</b>
26.1	访问者模式概述 .....	380
26.2	访问者模式的结构与实现 .....	382
26.2.1	访问者模式的结构 .....	382
26.2.2	访问者模式的实现 .....	383
26.3	访问者模式的应用实例 .....	385
26.4	访问者模式与组合模式联用 .....	393
26.5	访问者模式的优缺点与适用环境 .....	393
26.5.1	访问者模式的优点 .....	394
26.5.2	访问者模式的缺点 .....	394
26.5.3	访问者模式的适用环境 .....	394
26.6	本章小结 .....	394
26.7	习题 .....	395

附录 A UML 类图 .....	397
A.1 UML 概述 .....	397
A.2 类与类的 UML 表示 .....	398
A.3 类之间的关系 .....	400
附录 B 设计模式模拟试题 .....	408
B.1 模拟试题一 .....	408
B.2 模拟试题二 .....	411
B.3 模拟试题三 .....	414
参考文献 .....	417



# 第1章

## 设计模式概述

### 本章导学

随着面向对象技术的发展和广泛应用,设计模式已逐渐成为系统架构人员、设计人员、分析人员、维护人员及实现系统的一线程序员所需掌握的基本技能之一。设计模式广泛应用于面向对象系统的设计和开发,已成为面向对象技术的一个重要组成部分。当人们在特定的环境下遇到特定类型的问题时,可以采用他人已使用过的一些成功的解决方案,一方面降低了分析、设计和实现的难度,另一方面可以使系统具有更好的可维护性和可复用性。

本章将学习设计模式的定义、基本要素和分类,以及 GoF 的 23 种设计模式和设计模式的优点。

### 本章知识点

- 设计模式的诞生与发展。
- 设计模式的定义。
- 设计模式的基本要素。
- 设计模式的分类。
- 设计模式的优点。

## 1.1 设计模式的诞生与发展

与很多其他软件工程技术一样,设计模式起源于建筑领域,它是对前人经验的总结,为后人设计与开发基于面向对象的软件提供指导方针和成熟的解决方案。

### 1.1.1 模式的诞生与定义

模式(Pattern)起源于建筑业而非软件业,模式之父——美国加利福尼亚大学环境结构中心研究所所长 Christopher Alexander 博士用了约 20 年的时间,对舒适住宅和周边环境进行了大量的调查和资料收集工作,发现人们对舒适住宅和城市环境存在一些共同的认同