

SAMS

C#

Primer Plus

中文版

人民邮电出版社  
POSTS & TELECOMMUNICATIONS PRESS

[美] Klaus Michelsen 著  
云巅工作室 译

# C# Primer Plus 中文版

[美] Klaus Michelsen 著

云巅工作室 译

人民邮电出版社

## 图书在版编目 (CIP) 数据

C# Primer Plus 中文版 / (美) 米切尔森 (Michelsen, K.) 著; 云巅工作室编译.  
—北京: 人民邮电出版社, 2002.7

ISBN 7-115-10052-7

I. C... II. ①米... ②云... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 029534 号

## 版权声明

Klaus Michelsen: C# Primer Plus  
Copyright © 2002 by Sams Publishing  
Authorized translation from the English language edition published by the Sams Publishing.  
All rights reserved.

本书中文简体字版由美国 Sams 出版公司授权人民邮电出版社出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

## C# Primer Plus 中文版

- 
- ◆ 著 [美] Klaus Michelsen  
译 云巅工作室  
责任编辑 陈冀康
  
  - ◆ 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>  
读者热线 010-67180876  
北京汉魂图文设计有限公司制作  
北京顺义振华印刷厂印刷  
新华书店总店北京发行所经销
  
  - ◆ 开本: 787×1092 1/16  
印张: 44.25  
字数: 1 435 千字 2002 年 7 月第 1 版  
印数: 1-4 000 册 2002 年 7 月北京第 1 次印刷

著作权合同登记 图字: 01 - 2001 - 4088 号

ISBN 7-115-10052-7/TP · 2737

定价: 68.00 元

本书如有印装质量问题, 请与本社联系 电话: (010)67129223

## 内 容 提 要

本书全面介绍了微软最新推出的编程语言 C#。第 1 章介绍了学习 C#必需的软、硬件概念。第 2~5 章介绍了 C#的基本语言元素与结构。第 6~9 章讲解了进行 C#编程的一些概念,包括类型、操作符、分支语句、迭代语句等。第 10~11 章讨论了数组的使用。第 12~18 章详细讨论了如何运用 C#进行面向对象编程。第 19~20 章分别介绍了 C#中的异常处理、代表与事件。第 21~23 章介绍了一些 C#编程的相关知识,如提取 XML 说明文档、I/O 基础知识及递归。每章都配备了大量复习题和编程练习,附录部分提供了参考答案。

本书面向 C#初学者也适合于有经验的程序员学习和转向 C#编程。

## 译者的话

比尔·盖茨在展望微软的技术发展时曾提出“未来5年，我们的目标就是超越今天各自为营的Web站点，把Internet建成一个可以相互交换组件的地方。”2000年6月22日，.NET的推出正是微软实现这一计划的关键一步，微软宣称“其意义不亚于从DOS改变为Windows的过程”。.NET的含义深远，但简而言之，.NET是一种面向网络、支持各种用户终端的开发平台。2002年2月正式推出的Visual Studio.NET更是实现微软整个.NET战略的重型武器。尤其引人注目的是，在这“重型武器”之中还隐藏了另一种强大的武器——这就是最新诞生的C#。

如果你是（或即将成为）基于.NET平台的专业开发人员，C#开发语言将是你的最佳选择，因为：

- 它能与.NET平台无缝连接，充分利用.NET的先进性；
- 它结合了Visual Basic的高效和C++质朴的力量。

另外，C#作为微软全新推出的一种面向对象开发语言，它可以让开发人员快速地构建从底层系统到高层商业组件的不同级别的应用。

C#的优秀性能必然使更多的程序员投入它的怀抱。本书正是为C#初学者准备的一本知识全面的参考书籍。

《C# Primer Plus 中文版》一书为初学者量身定做。本书讲解知识全面，内容安排合理。本书不仅系统讲解了C#的主要语言元素、面向对象编程所必需的类、对象、继承、多态等高级概念，还介绍了C#引入的一些新概念，如代表、封装、接口等。

让我们以此书为起点，开始充分体验C#中“#”——sharp——的真正内涵吧！

本书由云巅工作室组织翻译，周良忠博士负责统稿。

由于译者水平有限，错误在所难免，望广大读者不吝指正。云巅工作室联系 E-mail: webmaster@CloudCrown.com。

云巅工作室  
2002年3月

# 前 言

## 原理、目标和读者

C#像其他编程语言一样，它也包含一套基本的语言元素。要形成有效的C#“句子”，必须根据C#的语法规则将这些元素放在一起。与创建成功、强大和完善程序中的一组C#“句子”相比，这也显得容易些。

本书不仅讲解了C#语言元素的基础、最佳用途，还教授如何将它们组合成功能强大的程序。后者介绍一些经过长期检验的基本技术（重点是面向对象编程）。为了实现这两个目标，而且避免分散你的学习精力，这里仅讨论了与C#直接相关的.NET部分内容。因而，本书不是关于.NET的ASP+、Web服务或Win Forms的书，而是一本旨在让你成为熟练的C#程序员的书。

本书面向初学者，学习过程中不需要数学技能，只需要一些简单的代数知识。真正的初学者应该逐页阅读此书。

如果你对另一种编程语言有一定了解，你也可以从本书中受益。在这种情况下，也许你想着重了解第1章和第2章中与.NET及C#相关的部分，而略过第3章对抽象和封装的讨论（如果你学习过面向对象编程）。

依我看，结合几种不同的教学工具学习一种新的编程语言会取得最佳的学习效果，一个知识面可以从多个不同的角度来了解和掌握。所以，本书不仅包括对不同C#元素的基本描述，而且还包括大量图示、源代码例子（附带有输出样例及分析）、实例学习、注释、技巧、常见错误、语法箱以及复习题（在附录A“小测验和练习答案”中提供了答案）。

学习编程不是背一些干巴巴的理论和概念，认识这一点很重要。相反，编程是一个实践、排错、不断提高创造性以及从中获取乐趣的过程。所以，每一章后面均提供了一套编程练习，你可以通过它们来练习、测试和提高编程技巧。

## 本书是如何组织的

本书分为23章和6个附录。其中的5个附录（附录B~F）放在SAMS网站（[www.sampublishing.com](http://www.sampublishing.com)）上。本书原书各为*C# Primer Plus*，原书ISBN为0-672-32152-1，你可根据这些信息在网站中查找相关资料。

第1章“计算机和计算机编程：基本概念”首先简要介绍了学习C#和计算机编程所需的计算机硬件和软件基本概念。本章的最后一部分介绍了.NET，并着重介绍了这一C#编程基本平台所提供的重要服务。

第2章“你的第一个C#程序”介绍了成功程序员开发一个健壮的计算机程序应遵循的基本软件开发过程，从中你可以初步接触面向对象编程的思想。本章最后演示了编写、编译和运行一个简单C#程序的方法。

通常，要了解一个初次造访的不熟悉地方，有效且易行的办法是找一个导游。第3章“C#导游：第一部分”及第4章“C#导游：第二部分”介绍了编写简单C#程序所需的C#语言主要知识。这里不是灌输一堆细节和理论，这两章的目的是让读者具备编写简单C#程序的能力并开始体验用C#进行编程。这两章通过几个简单的C#程序以直接明了的形式将这些知识娓娓道来。

第5章“你的第一个面向对象C#程序”分两部分，第一部分在读者从第3章和第4章获得了实践经验的基础上，进一步讲解C#语言的基本结构。第二部分详细讲解目前所获得的面向对象编程知识，并运用这些知识编写一段面向对象的电梯模拟程序。

现在，你知道如何编写一个简单的面向对象的C#程序了。但是，为了便于你顺利学习前面章节，很多C#语言知识诸如可能让你感到晦涩难懂的元素被刻意省略了。第6章“类型第一部分：简单类型”及后面的几个章节将着重拓宽你在前面5章所学的知识面。通过对C#中许多令人激动而且功能强大的元素进行系统、全面讨论来达到此目的。因此，你可以消化和吸收这些知识，并在自己的程序中付之实践。

第6章一般性地讨论了类型的概念及简单类型,尤其是大部分C#程序中所使用的简单类型。同时还介绍了一种在本书其余内容中用于描述不同C#元素语法的重要标记形式。

通过类型可以指定保存在程序中的数据属性。操作符一般作用于这些数据。没有操作符,程序数据就一成不变,大部分情况下也就无用武之地。第7章“类型第二部分:操作符、枚举器和字符串”对C#中不同种类的操作符进行了总体介绍,并详细讨论了主要的算术操作符。而且,本章讨论了派生类型——字符串(用于表示文本)及枚举(用于处理不变值)。本章提供并讨论了几个有用且有趣的字符串相关例程。

分支语句允许程序对不同的情况作出聪明的反应动作。第8章“流程控制第一部分:分支语句及相关概念”对C#中的整套分支语句结构进行了详细讨论,包括if、if-else和switch语句,同时讨论了与之密切相关的比较和逻辑操作符,它们也是实现第9章中迭代语句的关键元素。

迭代语句用于重复执行相同的动作,直到某条件不再满足为止。第9章“流程控制第二部分:迭代语句”讨论了C#的迭代语句,包括while循环、do-while循环及for循环。通过介绍嵌套迭代语句进一步讨论了迭代知识。最后,本章介绍了结构化编程概念,其中包含一些经过验证的构建健壮软件的设计原理。

数组用于表示小型或大型成组相似的数据项目。第10章“数组第一部分:数组基本知识”介绍了数组的概念,并讲解了如何使用第9章学过的迭代语句来有效访问它。本章还介绍了主要用于访问数组和其他数据项目集的foreach迭代语句。数组是一种引用类型——这是一个重要的面向对象概念。因此,本章以数组为例子来教授引用类型。

第10章只介绍了一维数组,它用于保存可以逐个排列的数据元素。有时,数据用二维表来表示更合适。第11章“数组第二部分:多维数组——数组检索与排序”就介绍了如何运用二维、三维或更多维的数组来表示这些类型的数据。多维数组与嵌套循环间的共生关系是数据结构与程序逻辑密切相关的很好例子。本章以几个著名的数组排序和检索算法结束,还举例说明如何通过重用.NET中的预构建程序部分来大量节省编程时间。

类和对象是面向对象编程的核心。类由程序员在源代码中编写,它充当程序运行时生成对象的蓝图。对象间相互使用来提供C#程序的功能。第12章“类解剖第一部分:静态类成员及方法的使用”先对类的组成作了简要介绍。后面内容与第13章和第14章一样对类的每个组成元素作了详细介绍。本章最先介绍的是属于某个类而不是任何特定对象的静态类成员。接着详细讨论了方法重载——这一机制可用于编写同名但不同接口及实现过程的不同版本的方法。

第13章“类解剖第二部分:对象创建和垃圾回收”介绍了如何初始化一个新对象、废除一个无用的旧对象。后者通过内置的垃圾回收机制来完成,它自动释放被无用对象占用的内存(用于程序新创建的对象)。

属性、索引器和用户自定义操作符重载是第14章“类解剖第三部分:编写直观代码”所介绍的C#元素。如果使用恰当,它们可以使源代码更简单、更直观。

名字空间有助于我们编写源代码时组织类,还可以让重用类的程序员更容易访问它们。第15章讨论了名字空间、编译单元和集合,还演示了它们之间不同组合的灵活性,以及如何为适应许多不同的应用程序配置而定制它们的整体布局。

通过继承可以从类B派生出类A,因而让类A自动继承类B的特征。类A的专有特征可以人工在源代码中添加。继承是面向对象编程中的一个重要概念,它具有许多优点,这将在第16章“继承第一部分:基本概念”中详细讨论。

第17章“继承第二部分:抽象函数、多态及接口”讨论了多态以及程序处理复杂问题时,如何通过多态应用几个简单的接口在高度抽象级别上编程。通过继承和抽象函数可以实现多态。本章还讨论了接口;通过接口我们可以在一个或多个继承层次结构间实现多态。

第18章介绍了结构,它在少数情况下可以替代类来使用。

异常是程序中的一个非正常条件,执行此类程序将导致.NET生成一个特殊的异常对象。第19章介绍了异常处理及如何避免程序非用户友好方式地终止等知识。

通常,当你编写源代码时就必须决定程序将执行的动作。通过代表可以将这一决定延迟到程序运行时来做。它可以让程序更加动态,更适应于多种情况的出现。第20章“代表及事件”讨论了代表及其与事件的关系。通过事件驱动编程,代表形成了事件的基础,这是现代许多应用程序类型的重要部分,尤其对于

图形化用户接口 (GUI)。

第 21 章“预处理、XML 文档及特性”解释了如何通过预处理指令简便地在最终程序中排除部分源代码。当你想根据同一个源代码生成多种不同的应用程序版本时，它可以发挥作用。本章接着介绍了直接从源代码中提取用户插入的注释得到 XML 格式的说明文档。本章最后介绍了特性及其给 C# 代码元素添加附加声明性信息的功能。

文件用来在磁盘上永久保存数据。程序可以重复读取文件，也可以将文件通过网络发送到远程计算机。第 22 章“文件 I/O 基础”简短介绍了文件 I/O，I/O 是文件输入（文件流入程序）和文件输出（文件流出程序）的缩写。

当某方法要求执行自身时，此方法就称为递归方法。第 23 章“递归基础知识”介绍了递归方法的使用。从表面上看，递归表示一个周期性、不可能的无用逻辑。但是，通过本章的学习，你就会明白递归可用于解决重要的计算问题。

附录 A 是“复习题和编程练习答案”提供了每章后面复习题和编程练习的答案。

附录 B 是“操作符优先级”。

附录 C 是“C# 中的保留字”。

附录 D 是“数字系统”讨论了二进制、八进制和十六进制数字系统。

附录 E 是“Unicode 字符集”。

附录 F 是“在控制台窗口使用 DOS 命令”。

## 约定

为了帮助读者区分本书中的不同类型信息，我们使用了不同的文本样式和布局。

- 重要词汇和新术语用突出显示。

- 由计算机生成的屏幕显示文字为以下样式：

```
I'm your trusted computer
```

- 由用户输入的屏幕显示文字为如下样式：

```
Hello computer, are you there?
```

- 对于用户按下的特殊按键，如回车键，用 <enter> 表示。

- 代码示例中的代码为以下样式：

```
private static void Main()
```

- 行连续符号  $\blacktriangleright$  放在源代码行的开头，它表示前一行代码太长，前一行与  $\blacktriangleright$  后的代码应看作同一行。

## 因特网上的 C#

虽然 C# 是一种相对较新的编程语言，但在因特网上也可以找到一些有用的 C# 信息资源。我个人认为，以下 3 个资源尤其有用：

- <http://msdn.microsoft.com/net>：这是微软开发者网络 (MSDN) 站点的 .NET 部分。在这里，你可以找到有关 C# 和 .NET 的最新语言规范、文档及技术文章。

- <http://www.devx.com/dotnet/resources/>：此网站包含因特网上完整的 .NET 和 C# 资源列表。

- <http://discuss.develop.com/dotnet.html>：此讨论组服务对象非初学者，微软的一些 C# 和 .NET 主要设计者以及其他一些资深 C# 程序员经常光顾这里。当你学习完本书时，你可能会受益于这里的一些有趣而且有用的知识讨论。



# 目 录

<b>第 1 章 计算机和计算机编程：基本概念</b> .....	1
1.1 C#是一种面向对象编程语言.....	2
1.2 硬件.....	3
1.2.1 主存储器的实质.....	4
1.2.2 数字系统.....	5
1.2.3 字节.....	5
1.2.4 文件.....	6
1.3 软件.....	6
1.3.1 执行 C#程序的大致过程.....	7
1.3.2 操作系统.....	7
1.3.3 编程语言和编译器.....	7
1.3.4 .NET 简介.....	8
1.3.5 在.NET 中编译 C#源代码.....	9
1.4 小结.....	10
1.5 复习题.....	11
<b>第 2 章 你的第一个 C#程序</b> .....	12
2.1 软件开发.....	13
2.2 软件开发过程.....	13
2.3 算法和伪码.....	14
2.4 3 个不同的错误类型.....	15
2.5 面向对象编程：初次体验.....	16
2.5.1 面向过程编程及其先天缺陷.....	16
2.5.2 面向对象编程及其优点.....	17
2.6 软件重用.....	19
2.7 .NET 中代码重用的基本单元：集合.....	21
2.7.1 从开发者角度看集合.....	21
2.7.2 从用户角度看集合.....	22
2.8 .NET 框架类库.....	23
2.9 C#的历史和设计目标.....	24
2.10 C#背景.....	24
2.11 C#的设计目标.....	26
2.12 能用 C#编写什么样的程序.....	27
2.13 创建 C#程序的原理.....	28
2.14 开始编程之前.....	28
2.15 选择文本编辑器.....	28
2.16 编写一个简单 C#程序的 7 步.....	29
2.16.1 打开和使用命令控制台（第 1 步）.....	29
2.16.2 输入和保存 C#源代码（第 2 步）.....	30
2.16.3 将源代码转换成一个 PE (.exe) 文件（第 3 步）.....	32
2.16.4 编译和查找错误（第 4 步）.....	32

2.16.5	运行程序 (第 5 步)	33
2.16.6	验证输入 (第 6 步)	33
2.16.7	庆祝的时刻 (第 7 步)	33
2.17	源代码简短分析	33
2.18	语法错误与编译器的注释	34
2.19	小结	34
2.20	复习题	35
2.21	编程练习	35
<b>第 3 章</b>	<b>C#导游: 第一部分</b>	<b>36</b>
3.1	简介	37
3.2	抽象和封装	37
3.2.1	抽象 (Abstraction)	37
3.2.2	封装 (Encapsulation)	38
3.2.3	学习面向对象编程的一个说明	41
3.3	一个交互式的 Hello World! 程序	42
3.3.1	编写 Hello.cs	42
3.3.2	Hello.cs 程序的基本元素	44
3.3.3	几个基本知识	51
3.4	小结	55
3.5	复习题	55
3.6	编程练习	56
<b>第 4 章</b>	<b>C#导游: 第二部分</b>	<b>57</b>
4.1	简介	58
4.2	SimpleCalculator.cs 程序所需元素	58
4.2.1	SimpleCalculator.cs 程序清单	58
4.2.2	对 SimpleCalculator.cs 程序的详细分析	60
4.3	用方法简化你的代码	67
4.4	方法就是构造块: 用关键字 private 封装你的 Helper 方法	68
4.5	小结	69
4.6	复习题	69
4.7	编程练习	70
<b>第 5 章</b>	<b>你的第一个面向对象 C# 程序</b>	<b>71</b>
5.1	简介	72
5.2	词汇结构	72
5.2.1	标识符和 CaPitalLcaTioN 样式	72
5.2.2	字面值	72
5.2.3	注释和代码注释	73
5.2.4	分隔符	73
5.2.5	操作符	73
5.2.6	关键字	74
5.3	对电梯模拟的一些思考	74
5.4	概念、目标和电梯模拟程序方案: 为评价电梯系统搜集有用统计数据	74

5.5 面向对象程序的一个实例 .....	75
5.5.1 程序 SimpleElevatorSimulation.cs 清单 .....	76
5.5.2 程序的整体结构 .....	78
5.5.3 对 SimpleElevatorSimulation.cs 程序的深入分析 .....	79
5.5.4 类关系和统一模型语言 (UML) .....	85
5.6 小结 .....	88
5.7 复习题 .....	88
5.8 编程练习 .....	89
<b>第 6 章 类型第一部分：简单类型</b> .....	<b>90</b>
6.1 简介 .....	91
6.2 概览 C# 中的类型 .....	91
6.2.1 什么是类型 .....	91
6.2.2 C#: 一个强类型化语言 .....	92
6.2.3 类型的优点 .....	93
6.2.4 C# 类型 .....	94
6.3 简单类型 .....	98
6.3.1 概观简单类型 .....	99
6.3.2 整数类型 .....	102
6.4 Blipos 时钟 .....	112
6.4.1 浮点类型 .....	119
6.4.2 decimal 类型 .....	123
6.4.3 浮点、decimal 和整数值的兼容性 .....	124
6.4.4 显式类型转换 .....	124
6.4.5 常量：字面值的符号名字 .....	125
6.4.6 格式化数字值 .....	128
6.4.7 bool 类型：初步接触 .....	130
6.5 小结 .....	130
6.6 复习题 .....	131
6.7 编程练习 .....	132
<b>第 7 章 类型第二部分：操作符、枚举器和字符串</b> .....	<b>133</b>
7.1 简介 .....	134
7.2 算术操作符及数字表达式 .....	134
7.3 结合性 .....	137
7.4 括号和优先级 .....	138
7.5 模操作符 (%) .....	139
7.6 一元操作符 .....	143
7.6.1 一元加和一元减 .....	143
7.6.2 自增和自减操作符 .....	144
7.7 判断表达式的类型 .....	146
7.8 在一个表达式中组合不同的类型 .....	148
7.9 访问组件的元数据：简介 .....	151
7.10 枚举常量 .....	154
7.11 枚举器操作符 .....	156

7.12	转换	157
7.13	System.Enum 方法	157
7.14	字符和文本	157
7.15	char 类型	157
7.16	string 类型	160
7.16.1	string 字面值和 string 对象	161
7.16.2	逐字符	161
7.16.3	使用字符串	162
7.16.4	在字符串中嵌入格式化数字	164
7.16.5	使用字符串	167
7.17	小结	178
7.18	复习题	179
7.19	编程练习	180
<b>第 8 章</b>	<b>流程控制第一部分：分支语句及相关概念</b>	<b>181</b>
8.1	流程控制简介	182
8.2	if 分支语句	183
8.2.1	简单 if 语句	183
8.2.2	复合语句	184
8.2.3	可选 else 子句	185
8.3	比较操作符和布尔表达式	187
8.4	嵌套 if 语句	189
8.5	多分支 if-else 语句	193
8.6	逻辑操作符	196
8.6.1	逻辑与 (AND) 操作符: &&	198
8.6.2	逻辑或 (OR) 操作符:	200
8.6.3	短路求值和位操作符&与	202
8.6.4	位异或操作符: ^	203
8.6.5	逻辑非 (NOT) 操作符: !	203
8.7	变量的作用域	205
8.8	goto 语句	208
8.9	switch 语句	210
8.9.1	switch 语句规则 1	213
8.9.2	switch 语句规则 2	214
8.9.3	switch 语句规则 3	216
8.9.4	switch 语句规则 4	218
8.9.5	使用 switch 语句	218
8.10	条件操作符	219
8.11	小结	219
8.12	复习题	220
8.13	编程练习	221
<b>第 9 章</b>	<b>流程控制第二部分：迭代语句</b>	<b>222</b>
9.1	遍历、分析和生成数据序列	223
9.2	while 循环语句	223

9.3 do-while 循环语句	227
9.4 for 循环语句	230
9.5 跳转语句 break 和 continue	235
9.5.1 break 语句	235
9.5.2 continue 语句	236
9.6 结构化编程和结构化的结构	237
9.7 组合赋值操作符	239
9.8 嵌套迭代语句	241
9.9 编程练习	246
9.10 小结	247
9.11 复习题	248
9.12 编程练习	249
<b>第 10 章 数组第一部分：数组基本知识</b>	<b>251</b>
10.1 数组的声明和定义	252
10.2 访问单个数组元素	255
10.3 数组索引越界	259
10.4 调整基于 0 的数组索引	262
10.5 初始化数组	263
10.6 用 foreach 语句遍历整个数组	264
10.7 System.Array 是一个引用类型	265
10.8 数组和相等性	267
10.9 数组和方法	269
10.9.1 数组元素作为方法中的参数	269
10.9.2 数组引用作为方法参数	270
10.9.3 克隆数组对象	273
10.10 执行数组值相等性比较的方法	276
10.11 命令行参数	277
10.12 使用数组作为方法的返回值	279
10.13 引用对象的数组元素	282
10.14 数组作为类中的实例变量	285
10.15 小结	292
10.16 复习题	293
10.17 编程练习	294
<b>第 11 章 数组第二部分：多维数组——数组检索与排序</b>	<b>295</b>
11.1 多维数组	296
11.1.1 二维数组	296
11.1.2 锯齿数组	311
11.2 多于二维的数组	314
11.3 System.Array 的内置方法	317
11.4 特殊数组技术	318
11.5 检索	325
11.5.1 顺序检索 (Sequential Search)	326
11.5.2 二分检索 (Binary Search)	327

11.5.3 用 System.Array 的 IndexOf 方法检索 .....	332
11.6 小结 .....	333
11.7 复习题 .....	333
11.8 编程练习 .....	334
<b>第 12 章 类解剖第一部分：静态类成员和方法的使用</b> .....	<b>335</b>
12.1 类解剖：概览 .....	336
12.2 数据成员 (data member) .....	338
12.2.1 实例变量 (instance variable) .....	338
12.2.2 静态变量 (static variable) .....	338
12.2.3 常量成员 (constant member) .....	344
12.2.4 只读成员 (readonly member) .....	345
12.2.5 声明数据成员：概览 .....	345
12.3 函数成员 .....	346
12.3.1 方法 .....	346
12.3.2 静态方法 (static methods) .....	348
12.4 小结 .....	370
12.5 复习题 .....	371
12.6 编程练习 .....	373
<b>第 13 章 类解剖第二部分：对象创建和垃圾回收</b> .....	<b>375</b>
13.1 实例构造函数 .....	376
13.1.1 为什么需要实例构造函数 .....	376
13.1.2 使用实例构造函数 .....	378
13.1.3 重载实例构造函数 .....	381
13.1.4 构造函数初始化器 .....	386
13.1.5 私有实例构造函数 .....	388
13.2 静态构造函数 .....	388
13.3 只读成员 (readonly member) .....	389
13.4 垃圾回收：自动动态内存管理 .....	390
13.4.1 对象如何被废弃 .....	390
13.4.2 垃圾回收器的任务 .....	393
13.4.3 释放非内存有限资源 .....	394
13.5 小结 .....	405
13.6 复习题 .....	406
13.7 编程练习 .....	407
<b>第 14 章 类解剖第三部分：编写直观代码</b> .....	<b>409</b>
14.1 属性 .....	410
14.1.1 属性与存取器和变异器方法 .....	410
14.1.2 使用属性提高效率 .....	415
14.1.3 用属性实现延迟初始化和惰性更新 .....	416
14.2 索引器：像数组一样使用对象 .....	419
14.2.1 从索引器所寄居的对象中调用索引器 .....	422
14.2.2 索引器重载：同一类中的多索引器 .....	422

14.2.3 避免滥用索引器 .....	426
14.3 操作符重载 .....	427
14.4 用户自定义隐式和显式转换 .....	434
14.4.1 需要用户自定义转换的两种情形 .....	435
14.4.2 使用非用户自定义转换技术 .....	436
14.4.3 用户自定义转换语法 .....	436
14.4.4 综合运用用户自定义和隐式转换 .....	441
14.5 嵌套类型 .....	442
14.5.1 嵌套类的优点 .....	443
14.5.2 简单嵌套类示例 .....	443
14.6 小结 .....	444
14.7 复习题 .....	444
14.8 编程练习 .....	446
<b>第 15 章 名字空间、编译单元和集合 .....</b>	<b>447</b>
15.1 自定义名字空间 .....	448
15.2 全局匿名名字空间 .....	448
15.3 名字空间和编译单元 .....	449
15.4 嵌套名字空间 .....	450
15.5 名字空间语法 .....	452
15.6 using 指示符的更多知识 .....	453
15.7 编译单元、名字空间及集合 .....	455
15.7.1 将几个编译单元编译成一个集合 .....	455
15.7.2 重用集合中的名字空间 .....	458
15.7.3 将名字空间分布到几个集合中 .....	460
15.8 应用 Ildasm 探索集合 .....	462
15.9 小结 .....	464
15.10 复习题 .....	464
15.11 编程练习 .....	466
<b>第 16 章 继承第一部分：基本概念 .....</b>	<b>467</b>
16.1 继承的必要性 .....	468
16.2 继承基础知识 .....	473
16.3 访问修饰符及继承 .....	480
16.3.1 访问修饰符 protected .....	480
16.3.2 访问私有基类成员 .....	480
16.3.3 internal protected 访问修饰符 .....	481
16.3.4 C# 的访问修饰符：概览 .....	481
16.4 派生类构造函数 .....	482
16.5 索引器也能被继承和覆盖 .....	485
16.6 在基类中调用覆盖函数 .....	487
16.7 通过继承重用 .NET 框架类库 .....	490
16.8 多级派生类 .....	491
16.9 方法覆盖及重载原理不同 .....	494
16.10 小结 .....	495

16.11 复习题 .....	496
16.12 编程练习 .....	497
<b>第 17 章 继承第二部分：抽象函数、多态及接口 .....</b>	<b>499</b>
17.1 抽象方法、属性、索引器及类 .....	500
17.2 多态 (polymorphism) .....	503
17.2.1 后代类的对象具有多个类型 .....	503
17.2.2 动态捆绑虚方法和 (get,set) 存取器 .....	503
17.2.3 实例学习：使用多态创建简单绘图程序 .....	505
17.3 丢失和重新获取类型信息 .....	511
17.3.1 is 操作符 .....	511
17.3.2 对象类型转换 .....	511
17.3.3 as 操作符 .....	514
17.4 最终的基类：System.Object .....	514
17.5 隐藏方法 .....	518
17.6 用 new 和 override 关键字控制版本 .....	520
17.7 多重继承 .....	522
17.8 接口 .....	523
17.8.1 定义接口 .....	524
17.8.2 实现接口 .....	526
17.8.3 应用接口的一般编程 .....	528
17.8.4 构建接口层次 .....	531
17.8.5 接口转换 .....	532
17.8.6 覆盖虚接口 .....	532
17.8.7 显式实现接口函数 .....	533
17.9 小结 .....	535
17.10 复习题 .....	535
17.11 编程练习 .....	536
<b>第 18 章 结构 .....</b>	<b>538</b>
18.1 定义结构 .....	539
18.2 装箱和拆箱 .....	541
18.3 用与不用 new 关键字创建结构 .....	542
18.4 值类型和引用类型 .....	543
18.5 小结 .....	544
18.6 复习题 .....	544
18.7 编程练习 .....	544
<b>第 19 章 异常处理 .....</b>	<b>545</b>
19.1 异常处理概览 .....	546
19.2 没有 try-catch-finally 的情形 .....	546
19.3 try 和 catch 块 .....	548
19.4 捕获函数调用链上端的异常对象 .....	550
19.5 多重 catch 块 .....	551
19.6 finally 块 .....	552



19.7	嵌套 try 块	554
19.8	用 throw 显式抛出异常	555
19.9	编写自定义异常	557
19.10	小结	560
19.11	复习题	560
19.12	编程练习	561
<b>第 20 章</b>	<b>代表和事件</b>	<b>562</b>
20.1	代表	563
20.1.1	代表数组及方法调用中作为参数的代表	565
20.1.2	组播代表	569
20.2	事件	571
20.3	小结	577
20.4	复习题	577
20.5	编程练习	577
<b>第 21 章</b>	<b>预处理、XML 文档及特性</b>	<b>579</b>
21.1	预处理程序指令	580
21.1.1	用 #define、#if 和 #endif 排除和包含代码	580
21.1.2	#undef: 取消定义标识符	582
21.1.3	#elif 和 #else	582
21.1.4	#error 和 #warning	582
21.1.5	#region 和 #endregion	582
21.1.6	#line	582
21.2	XML 说明文档	583
21.3	特性	584
21.4	小结	588
21.5	复习题	588
21.6	编程练习	589
<b>第 22 章</b>	<b>文件 I/O 基础</b>	<b>592</b>
22.1	对文件的需要	593
22.2	流基础和文件 I/O	593
22.3	文件 I/O 类概览	593
22.4	FileInfo 类	594
22.5	用 StreamReader 和 StreamWriter 进行文本文件输入和输出	596
22.6	用 FileStream 类进行二进制输入和输出	600
22.7	小结	602
22.8	复习题	602
22.9	编程练习	603
<b>第 23 章</b>	<b>递归基础知识</b>	<b>604</b>
23.1	不同方法的待决方法实例	605
23.2	同一个方法的待决方法实例	606
23.3	应用递归计算 n 的阶乘	608