

高职高专计算机教学改革 新体系 规划教材

C#程序设计 案例教程

张志强 主 编

李小强 孙玉敏 副主编

清华大学出版社



C#程序设计案例教程

张志强 主 编

李小强 孙玉敏 副主编

清华大学出版社
北 京

内 容 简 介

本书吸纳了作者多年来教学科研成果和 C# 项目开发经验,采用案例驱动方式组织教学内容。全书共 12 章,重点讲述了.NET 知识、C# 程序设计基础、面向对象程序设计、Windows 应用程序设计、ADO.NET 数据库访问技术、集合与泛型、文件管理、软件部署与安装等内容。全书 200 余个从各类经典项目中提取的精品案例,融于 C# 各章节的知识点中,并通过精品案例、综合案例、技能测试、项目实训等多个教学环节展现。同时,对每个案例的重点和难点代码加以注释,并以编程知识、编程经验、编程提示等形式进行强调和扩展,以求达到编程知识和项目开发实践经验的融会贯通。

本书内容翔实,案例丰富,图文并茂,实用面广,可操作性强,项目开发实践特色鲜明。既可作为各类高职院校和成人高校相关专业的 C# 教学用书或行业技术人员的培训教材,也可作为计算机应用人员和计算机编程爱好者的参考书。

为了便于读者分析和测试书中的示例,随书提供了所有精品案例、综合案例的源代码和案例的素材库;同时,为便于教师进行多媒体课堂教学,还提供了配套的 PPT 电子教案、教学计划等教学资源,可登录 www.tup.com.cn 免费下载。

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# 程序设计案例教程/张志强主编.--北京:清华大学出版社,2013

高职高专计算机教学改革新体系规划教材

ISBN 978-7-302-32677-9

I. ①C… II. ①张… III. ①C 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 122395 号

责任编辑:陈砺川

封面设计:傅瑞学

责任校对:袁芳

责任印制:宋林

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

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795764

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:15.5

字 数:359 千字

版 次:2013 年 8 月第 1 版

印 次:2013 年 8 月第 1 次印刷

印 数:1~3000

定 价:32.00 元

产品编号:052636-01

前言

FOREWORD

C# 结合 C/C++ 和 Java 等语言的特点,是基于优秀的 .NET 平台而设计的一种全新程序设计语言。它直接调用 .NET Framework 中完整且丰富的类库,可开发出跨平台的应用软件系统。C# 具有简单易学、功能强大、可视化设计等特点,已经成为业界主流程序设计语言。

本书编写力求体现先进性、实用性、适应性、通俗性和灵活性的特点。全书共分为 12 章,采用案例驱动组织教学内容。200 余个从各类经典项目中精心提取的精品案例,有针对性地融入 C# 各章节的知识点中,并通过综合案例、技能测试、项目实训等环节综合演练,以达到编程理论和项目经验的融会贯通。各章内容概括如下。

第 1 章为 .NET 和 C# 概述;第 2 章为 C# 编程基础;第 3 章为 C# 程序流程控制;第 4 章为数组和字符串;第 5 章为 C# 实现 OOP 基础;第 6 章为继承和多态;第 7 章为 Windows 窗体编程基础;第 8 章为深入 Windows 控件编程;第 9 章为 ADO.NET 访问数据库;第 10 章为集合与泛型;第 11 章为文件和数据流操作;第 12 章为 Windows 应用程序打包部署。各章以技能目标、案例任务、章节内容、精品案例、综合案例、本章总结、技能测试、项目实训等形式组织教学内容,并对难点和重点以代码注释、编程知识、编程经验、编程提示等形式强调。

书中实例丰富,所有精品案例、综合案例、项目实训等,均是经典项目的关键与归纳。通过本书的学习,可以加深对程序设计的逻辑基础的理解,并在短时间内轻松掌握 C# 项目开发知识与经验。

本书由河南职业技术学院张志强主编,李小强、孙玉敏任副主编。张志强编写第 1 章和第 9 章,孙玉敏编写第 2 章、第 6 章和第 10 章,李明、胡春春编写第 3 章,韩敏编写第 4 章,郑宝林编写第 5 章和第 11 章,娄松涛、张铁头、翟慧编写第 7 章,李小强编写第 8 章和第 12 章,全书由张志强统一修改和定稿。此外,在成书过程中还得到了清华大学出版社给予的支持和帮助,在此表示衷心的感谢。本书在编写过程中,参考了大量的著作和文献资料,在此一并向有关作者、编者表示真诚的感谢。

本书既可作为各类高等院校计算机及相关专业 C# 教学用书,贴近实战的翔实案例也可作为计算机应用项目开发人员和计算机编程爱好者的参考书。

书中介绍了如何安装和卸载 Visual Studio 2012 中文版,在实际教学中,书中提供的案例及项目实训内容可以在 Visual Studio 2010 环境下运行。由于时间和作者水平有限,书中难免出现差错和不妥之处,欢迎读者批评指正!

为了便于读者测试和分析书中的示例,随书提供了所有精品案例、综合案例的源程序、素材库;同时,为了便于教师进行多媒体课堂教学,随书还提供了相应的 PPT 电子教案、教学计划等教学资源,读者可至 www.tup.com.cn 免费下载。

编 者

2013 年 5 月

目 录

CONTENTS

第 1 章 .NET 和 C# 概述 /1

1.1 C# 语言	1
1.1.1 C# 的起源	1
1.1.2 C# 的应用领域	3
1.1.3 C# 的特点	4
1.2 Microsoft .NET Framework	5
1.2.1 .NET 简介	5
1.2.2 .NET Framework	5
1.2.3 C# 与 .NET 的关系	6
1.3 .NET 集成开发环境 Visual Studio	6
1.3.1 系统安装条件	6
1.3.2 Visual Studio 2012 的安装与卸载	7
1.3.3 Visual Studio 2012 开发环境简介	7
1.4 C# 编程体验	9
1.4.1 控制台应用程序	9
1.4.2 Windows 窗体应用程序	11
1.5 C# 程序结构	13
1.5.1 命名空间	13
1.5.2 类	14
1.5.3 Main() 方法	14
1.5.4 标识符与关键字	14
1.5.5 C# 语句	15
1.5.6 注释	15
1.6 本章总结	15
1.7 技能测试	16
1.8 项目实训	16

第 2 章 C# 编程基础 /17

2.1 变量和常量	17
-----------------	----

2.1.1	变量	17
2.1.2	常量	18
2.2	数据类型	19
2.2.1	值类型	19
2.2.2	引用类型	22
2.2.3	类型转换	22
2.3	表达式与运算符	26
2.3.1	表达式	26
2.3.2	运算符	27
2.3.3	运算符优先级	32
2.4	本章总结	33
2.5	技能测试	33
2.6	项目实训	34
第 3 章	C# 程序流程控制	/36
3.1	顺序结构	36
3.2	选择结构	37
3.2.1	if 语句	37
3.2.2	switch 语句	42
3.3	循环结构	44
3.3.1	while 语句	44
3.3.2	do...while 语句	45
3.3.3	for 语句	46
3.3.4	循环的嵌套	47
3.4	跳转语句	48
3.4.1	break 语句	48
3.4.2	continue 语句	49
3.4.3	goto 语句	50
3.4.4	return 语句	51
3.5	本章总结	51
3.6	技能测试	52
3.7	项目实训	52
第 4 章	数组和字符串	/55
4.1	数组概述	55
4.2	数组的声明与使用	56
4.2.1	一维数组的声明与使用	56
4.2.2	二维数组的声明与使用	57

4.3 数组的操作.....	60
4.3.1 数组的遍历	60
4.3.2 数组的查找	61
4.3.3 数组的排序	62
4.4 字符串.....	65
4.4.1 访问字符串的字符	65
4.4.2 常用字符串处理方法	65
4.4.3 格式化字符串的输出方法	67
4.5 本章总结.....	68
4.6 技能测试.....	68
4.7 项目实训.....	69

第 5 章 C# 实现 OOP 基础 /71

5.1 面向对象概述.....	71
5.2 类与对象.....	72
5.2.1 类的概念	72
5.2.2 类的声明	72
5.2.3 对象	73
5.2.4 类与对象的关系	73
5.3 类的成员.....	74
5.3.1 类成员的修饰符	74
5.3.2 字段	74
5.3.3 常量	75
5.3.4 属性	76
5.3.5 索引器	78
5.3.6 构造函数和析构函数	80
5.4 方法.....	83
5.4.1 方法的声明	83
5.4.2 方法中的参数类型	85
5.4.3 静态方法与实例方法	88
5.4.4 方法的重载	89
5.5 本章总结.....	90
5.6 技能测试.....	90
5.7 项目实训.....	90

第 6 章 继承和多态 /92

6.1 继承.....	92
6.1.1 继承的概念	92

6.1.2	派生类的构造函数	93
6.1.3	protected 成员	95
6.2	多态	96
6.2.1	虚拟方法和重写方法	96
6.2.2	抽象类和抽象方法	98
6.2.3	接口	100
6.2.4	密封类和密封方法	102
6.3	本章总结	103
6.4	技能测试	103
6.5	项目实训	103

第 7 章 Windows 窗体编程基础 /105

7.1	Windows 窗体程序开发步骤	105
7.1.1	项目建立	105
7.1.2	界面设计与属性设置	107
7.1.3	代码编写	109
7.1.4	运行调试	110
7.1.5	方案发布	110
7.2	Windows 窗体程序结构	110
7.3	Windows 窗体容器	111
7.3.1	窗体容器可视化操作	112
7.3.2	窗体容器属性设置	113
7.3.3	窗体事件	115
7.3.4	窗体可视化编程	116
7.4	Windows 窗体控件	117
7.4.1	控件概述	117
7.4.2	控件属性和事件	117
7.4.3	控件基本操作	119
7.5	常用窗体控件	120
7.5.1	标签控件	120
7.5.2	按钮控件	121
7.5.3	文本框控件	121
7.5.4	单选按钮控件	122
7.5.5	复选框控件	123
7.5.6	列表框控件	123
7.6	消息框窗口	123
7.7	综合案例：计算器设计	124
7.7.1	项目建立	125

7.7.2 项目界面设计	125
7.7.3 代码编写	126
7.7.4 运行调试	127
7.7.5 方案发布	127
7.7.6 问题讨论	128
7.8 本章总结	128
7.9 技能测试	128
7.10 项目实训	129

第 8 章 深入 Windows 控件编程 /132

8.1 窗体修饰控件	132
8.1.1 菜单控件	132
8.1.2 选项卡控件	134
8.1.3 工具栏控件	134
8.1.4 状态栏控件	135
8.1.5 分组框控件	135
8.2 ImageList 组件	136
8.2.1 添加图像	136
8.2.2 删除图像	137
8.3 ListView 控件	138
8.3.1 添加移除项	138
8.3.2 选择默认项	140
8.3.3 添加项图标	141
8.3.4 显示平铺视图	142
8.4 DateTimePicker 控件	143
8.4.1 显示日期	143
8.4.2 自定义格式显示日期	144
8.4.3 返回 DateTimePicker 控件中自定义日期	145
8.5 其他高级控件	146
8.5.1 Timer 控件	146
8.5.2 ProgressBar 控件	147
8.6 MDI 窗体	148
8.6.1 MDI 窗体概念	148
8.6.2 MDI 窗体设置	149
8.6.3 MDI 子窗体排列	149
8.7 继承窗体	152
8.7.1 继承窗体概念	152
8.7.2 继承窗体创建	152

8.7.3 继承控件属性	154
8.8 综合案例:投票支持系统	155
8.8.1 投票支持系统界面设计	155
8.8.2 代码编写	156
8.8.3 运行结果	161
8.8.4 问题讨论	162
8.9 本章总结	162
8.10 技能测试	162
8.11 项目实训	163
第9章 ADO.NET 访问数据库 /165	
9.1 数据库概述	165
9.1.1 数据库系统	165
9.1.2 数据库设计的基本步骤	166
9.1.3 结构化查询语言	166
9.2 ADO.NET 简介	170
9.3 Connection 对象	171
9.3.1 连接数据库	171
9.3.2 关闭连接	173
9.4 Command 对象	174
9.4.1 设置数据源类型	174
9.4.2 执行 SQL 语句	174
9.5 DataReader 对象	178
9.5.1 DataReader 对象概述	178
9.5.2 判断查询结果中是否有值	179
9.5.3 读取数据	180
9.6 DataAdapter 对象	181
9.6.1 DataAdapter 对象概述	181
9.6.2 填充 DataSet 数据集	182
9.6.3 更新 DataSet 数据集	182
9.7 DataSet 对象	183
9.7.1 DataSet 概述	183
9.7.2 DataSet 创建	183
9.7.3 DataSet 与 DataAdapter	183
9.8 DataGridView 控件	183
9.8.1 DataGridView 控件概述	183
9.8.2 DataGridView 控件显示数据	184
9.9 综合案例:超市商品查询系统	185

9.9.1 超市商品查询系统主界面设计	185
9.9.2 代码编写	186
9.9.3 运行结果	189
9.9.4 问题讨论	189
9.10 本章总结	189
9.11 技能测试	190
9.12 项目实训	191

第 10 章 集合与泛型 /194

10.1 集合概述	194
10.1.1 集合的定义	194
10.1.2 C# 集合类型	195
10.2 数组列表 ArrayList	195
10.2.1 ArrayList 定义	195
10.2.2 添加成员	195
10.2.3 删除成员	196
10.2.4 遍历成员	197
10.3 字典 Hashtable	198
10.3.1 Hashtable 定义	198
10.3.2 Hashtable 简单操作	198
10.3.3 Hashtable 遍历	199
10.4 泛型和泛型集合	200
10.4.1 泛型集合 List<T>	200
10.4.2 泛型集合 Dictionary<K,V>	202
10.5 本章总结	204
10.6 技能测试	205
10.7 项目实训	206

第 11 章 文件和数据流操作 /207

11.1 文件操作概述	207
11.1.1 System.IO	207
11.1.2 文件和流	208
11.1.3 File 类	208
11.1.4 FileInfo 类	210
11.2 文件基本操作	211
11.2.1 创建文件	211
11.2.2 打开文件	213
11.2.3 判断文件是否存在	213

11.2.4	读写文件	214
11.2.5	复制文件	215
11.2.6	移动文件	215
11.2.7	删除文件	215
11.3	综合案例: 文本读写器	215
11.4	乱码问题	217
11.5	文件夹基本操作	218
11.5.1	Directory 类	218
11.5.2	DirectoryInfo 类	220
11.6	本章总结	221
11.7	技能测试	221
11.8	项目实训	222

第 12 章 Windows 应用程序打包部署 /225

12.1	系统安装程序	225
12.2	Windows 安装项目的创建	226
12.3	Windows 安装项目制作	227
12.3.1	添加项目输出	227
12.3.2	添加内容文件	228
12.3.3	创建桌面快捷方式	229
12.3.4	添加注册表项	230
12.3.5	生成 Windows 安装程序	231
12.4	部署 Windows 应用程序	233
12.5	本章总结	234
12.6	技能测试	234
12.7	项目实训	234

参考文献 /236

.NET 和 C# 概述

Chapter 1

技能目标

- 了解 C# 语言
- 了解安装与配置 Visual Studio 2012
- 熟悉 Visual Studio 2012 集成开发环境
- 掌握创建和运行控制台应用程序的方法
- 掌握创建 Windows 窗体程序的方法和步骤

案例任务

- 实现简单控制台应用程序开发
- 实现简单 Windows 应用程序设计

1.1 C# 语言

1.1.1 C# 的起源

C#(读“C sharp”)是微软公司在 2000 年 7 月发布的一种全新且简单、安全、面向对象的程序设计语言,是专为基于 .NET 平台开发而量身打造的全新高级语言,是 .NET 开发人员的首选语言。它的功能经受住了时间的考验,始终位于科技前沿,为现代数字化环境开发提供了一种功能强大、性能高效的编程方法。无论从哪个角度看,C# 都是 21 世纪最重要的编程语言之一。

C# 语言是基于当时程序语言无法解决软件设计中的一些关键问题而创建的。当其他语言解决了可移植性问题时,却没有解决基于网络环境的现代软件工程管理和软件应用过程急需的问题。

首先是多语言互操作性,即混合语言程序设计问题,实现一种语言编写的代码和另一种语言编写的代码协同工作问题。多语言互操作性是现代创建分布式软件系统、分布式软件工程管理及维护所必需的功能,也是软件组件式开发、软件工程管理所期望的内容。只有多语言互操作,才能解决最有价值的组件被尽可能多的语言采用的问题,才能最广泛地应用于不同的网络操作环境,才能进行分布式软件开发与维护。

另外,由于 Windows 是当今应用最广泛的操作系统,其他的语言没有做到与之完全兼容。尽管有些语言以虚拟机等手段实现了在 Windows 操作平台的运行,但不能充分发挥软/硬件资源效能。

为满足软件开发与应用的种种需要,微软公司开发了 C# 语言,并且将其定位为整个 .NET 战略的一部分。C# 的首席设计师是在软件开发领域做出非凡贡献的安德斯·海尔斯伯格。2000 年中期发布了 C# 的第一个版本。到目前为止,已经发展到支持 .NET Framework 4.5 版本的强大语言开发工具。

C# 的族谱如图 1.1 所示。C# 的“祖父”是 C,从 C 那里继承了语法、许多关键字和运算符。接下来,C# 基于并改进了 C++ 所定义的对象模型。所以读者只要有一些计算机软件基础,学习 C# 语言将不会感到陌生。

C# 吸收了 C++、Visual Basic、Delphi、Java 等语言的优点,体现了当今最新的程序设计技术的功能和精华。C# 不仅继承了 C 语言的语法风格、C++ 的面向对象特性。而且,C# 的对象模型已经面向 Internet 进行了重新设计,基于 .NET Framework 的类库;C# 不再提供对指针类型的支持,使得程序不能随便访问内存地址空间,从而更加健壮;C# 不再支持多重继承,避免了以往类层次结构中由于多重继承带来的可怕后果。.NET Framework 为 C# 可持续发展提供了一个强大的、易用的、逻辑结构一致的程序设计环境。同时,公共语言运行时(Common Language Runtime)为 C# 程序语言提供了一个托管的运行环境,使程序比以往更加稳定、安全。

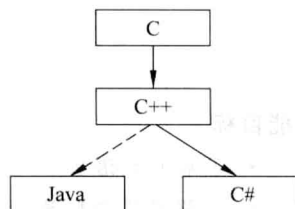


图 1.1 C# 族谱

编程知识

安德斯·海尔斯伯格(Anders Hejlsberg,1960—),丹麦人,Turbo Pascal 编译器的主要作者,Delphi 和 .NET 之父。

安德斯·海尔斯伯格曾在丹麦科技大学学习工程学,但没有毕业。他第一次登上软件界历史舞台是在 20 世纪 80 年代早期,是因为他为 MS-DOS 和 CP/M 设计了一个 Pascal 编译器。当时身为一个小公司的 Borland 很快雇佣了他,并买下了他的编译器,改称 Turbo Pascal。在 Borland,海尔斯伯格继续开发 Turbo Pascal,并最终带领他的小组设计了 Turbo Pascal 的替代品、开发工具史上的奇迹——Delphi。

1996 年,安德斯·海尔斯伯格在 Delphi 3 发布前以年薪 300 万美元的身价离开 Borland 转而加盟 Microsoft(微软)。据说,比尔·盖茨亲自参与了这次挖角行动,并许诺安德斯·海尔斯伯格在微软将得到技术上的足够自由和资源支持。该事件也是微软公司和 Borland 公司后续官司的导火索。

后来在微软公司把视窗操作系统和软件向网络迁移的新市场战略中,C# 语言是最重要的环节。微软动用了最好的资源,安德斯担任了 C# 语言的首席设计师,同时也是微软 .NET 战略构架的重要决策者。C# 完成软件开发方式重大变革,也赢得软件开发者的拥戴。

1.1.2 C# 的应用领域

C# 语言出身名门,并继承了优良的血统,保留了主流软件开发语言的优点,摒弃了它们的缺点,这使得 C# 语言更加完美,所以其应用领域也十分广泛,已应用到了软件开发行业中各个领域。

(1) 数据库应用开发领域

数据库应用程序,也就是企事业单位所使用的业务管理软件,C# 友好的设计界面,强大的语言实现方式,基于 ADO.NET 的数据库访问技术非常适合数据库应用程序开发。

(2) 分布式开发领域

分布式开发常用于设计比较大型的数据库应用程序,比如交通行业的售票系统、银行的金融管理系统、公安部门的全国户籍管理系统、电子商务管理系统等,C# 语言的 ADO.NET 技术、WCF 技术、Web Service 技术等非常适合企业分布式应用程序开发。

(3) Web 应用开发

采用 ASP.NET 技术开发 Web 应用程序十分流行,而 C# 正是该技术的主要后台开发语言。ASP.NET 代表网站有 Microsoft、HP、Dell 等。

(4) 网络通信开发

基于网络的各种即时通信软件在工作和生活中应用得非常普遍,比如 QQ 软件、迅雷下载工具等,C# 语言的多线程技术、数据流技术、Socket 通信技术等都非常适合网络通信开发。

(5) 嵌入式设备开发

基于 Windows CE 操作系统的智能手机、PDA 等时尚消费类电子产品应用非常广泛,其内部的应用软件、游戏等就是使用 C# 语言进行嵌入式开发的。

(6) 游戏软件开发

C# 语言通过与 XNA 游戏开发框架相结合,可以开发出效果非常绚丽的游戏。如动力时空开发的以中国经典武侠文化为背景的网络游戏《至尊天下》就是使用 C# 语言开发的。

(7) 工具软件或组件开发

使用计算机时,经常会用到工具软件、组件等,比如绘图工具、图片浏览器、播放软件等,C# 语言可以使用先进的 GDI+ 绘图技术、COM 组件技术等来实现。



编程知识

XNA 中的 X 表示能够在 Windows、Xbox 和合作伙伴之间达到跨平台的强大的软件工具。N 表示“下一代(Next-generation)”,A 表示“架构(Architecture)”。它只是一个游戏开发框架。XNA Studio 是 XNA 平台上的首选开发工具,而且将是 XNA 品牌中发布的第一个产品。基于 C# 的 XNA 能够:允许开发者将革新付诸于实践、有助于节省各式各样的开销、提供跨设备开发的机会、很容易创建一鸣惊人的游戏、使开发人员能够专注

于游戏设计,而在处理硬件复杂性上减少时间、提供真实的图像、声音和移动,允许开发者同步地设计一个游戏的 Windows 和 Xbox 版本。

1.1.3 C# 的特点

由于 C# 面向对象的卓越设计与性能,无论是构建高级的商业对象还是网络系统级的应用程序开发,C# 都是理想之选。

(1) 效率与安全性

C# 在设计时就考虑了软件开发与管理竞争性问题。它使开发者用更少的代码做更多的事,同时也不易出错。使开发者可以不断缩短开发周期,不断推出应用程序的新版本。

(2) 支持现有的网络编程新标准

新的应用程序开发模型意味着越来越多的解决方案依赖于新出现的网络标准,例如 HTML、XML、SOAP 等。现存的开发工具往往都是早于 Internet 出现的,或者是在网络孕育期即出现的,所以一般无法很好地支持最新的网络技术。

C# 能够在 Microsoft .NET 平台便捷地构建应用程序的扩展框架。使任何组件轻松转化为 XML 网络服务,通过 Internet 被任何操作系统上运行的任何程序调用。

(3) 消除大量程序错误

即使是专家级的程序员也常会犯一些最简单的小错误,比如忘了初始化变量。但往往正是这些小错误带来难以预料的问题,有些甚至需要很长时间来寻找和解决。一旦一个程序作为产品来使用,就算最简单的错误,纠正起来也可能要付出极其昂贵的代价。C# 消除大量程序错误并且提供了良好的纠错能力,使程序开发更具优势。

(4) 对版本的更新提供的内在支持降低了开发成本

更新软件组件是一项很容易出错的工作,因为代码的修改可能无意间改变原有程序的语义。为协助开发者进行这项工作,C# 为版本的更新提供内在的支持。例如,方法重载必须显式声明。这样可以防止编码错误,保证版本更新的灵活性。还有一个相关的特性就是对接口和接口继承的内在支持。这些特性使得 C# 可以开发复杂的框架并且随着时间不断发展更新它。

这些特性使得开发程序项目的后续版本的过程更加健壮,从而减少后续版本的开发成本。

(5) 扩展交互性

作为一种自动管理的、类型安全的环境,C# 适合于大多数企业应用程序。但实际的经验表明有些应用程序仍然需要一些底层的代码,要么是因为基于性能的考虑,要么是因为要与现有的应用程序接口兼容。由于 C# 的面向对象特性更为彻底,一切皆对象,不存在独立的函数,因此,更容易开发性能优异的扩展交互性软件。

总之,C# 增强了开发者的效率,也致力于消除编程中可能导致的错误,同时也保持了开发者所需要的强大性和灵活性。