

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

C#程序设计 简明教程

Beginning C# for Developer

陈佛敏 潘春华 吕洋波 编著

- 精选内容，加强应用，无需语言基础
- 注重实践，示例丰富，各章附上机指导
- 知识全面，面向实际，涵盖C#窗体/网络应用



精品系列



人民邮电出版社
POSTS & TELECOM PRESS

TP312/2864

2008

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

C#程序设计 简明教程

Beginning C# for Developer

陈佛敏 潘春华 吕洋波 编著



精品系列

人民邮电出版社

北京

图书在版编目(CIP)数据

C#程序设计简明教程 / 陈佛敏, 潘春华, 吕洋波编著.
北京: 人民邮电出版社, 2008.5
21 世纪高等学校计算机规划教材. 精品系列
ISBN 978-7-115-17830-5

I. C… II. ①陈…②潘…③吕… III. C 语言—程序设计—
高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 033280 号

内 容 提 要

本书循序渐进地介绍了 C#的基础知识和基本应用。全书分为 9 章, 内容包括 C#与.NET 框架、C#语法基础、面向对象设计、数组和集合、字符串和正则表达式、文件操作、数据库开发技术、Windows 窗体应用、Web 网络应用等。书中涉及的每一个知识要点, 都通过具体的实例加以阐述, 使读者更容易理解。

本书可作为普通高等院校计算机科学与技术、网络工程、软件工程等专业 C#相关课程的教材, 也适合 C#初学者及相关培训机构使用。

21 世纪高等学校计算机规划教材——精品系列

C#程序设计简明教程

-
- ◆ 编 著 陈佛敏 潘春华 吕洋波
责任编辑 蒋 亮
 - ◆ 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>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 15.25
字数: 396 千字 2008 年 5 月第 1 版
印数: 1—3 000 册 2008 年 5 月河北第 1 次印刷

ISBN 978-7-115- 17830-5/TP

定价: 25.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

计算机应用能力已经成为社会各行业最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材——精品系列”。

“21世纪高等学校计算机规划教材——精品系列”具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科类专业的学生，尽量符合目前普通高等教学计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论技术与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善，共促提高。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、中国林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业和科研单位的领导或技术人员的积极配合。在此，人民邮电出版社向他们表示衷心的感谢。

我们相信，“21世纪高等学校计算机规划教材——精品系列”一定能够为我国高等院校计算机课程教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

C# (C Sharp) 作为微软公司专为 .NET Framework 量身定做的编程语言, 在 Web 系统、Web Service 开发、桌面应用、类库及 COM 开发等多个领域, 都显示了强大的功能。对于普通高等院校计算机科学与技术、网络工程、软件工程等专业的学生, 掌握 C# 开发技术将有助于满足目前软件开发、系统集成、Web 开发等领域企业对于 C# 开发能力的要求, 提高就业阶段自身的竞争力; 与此同时, 对于从事应用软件开发的人员或者将要进入该领域的初学者, 掌握 C# 开发技术将有助于更快地完成易维护、高效率、运行稳定的系统的开发工作, 并在开发的过程中体会到工作的乐趣。

本书遵循由浅入深、循序渐进的学习规律。所以, 在开始本书的学习之前, 读者可以不必有其他语言的学习使用经验。如果同学们在前期学习了其他高级语言, 如 C/C++、Java 等, 则可以更快速地掌握本书的重点。

本书从细节到整体, 从内容编排到目录组织, 都力图合理。在每一小节的内容编排上, 首先给出清晰易懂的基本概念, 然后试图挖掘更为深层的设计、编程和部署思想, 最后通过独立而有趣的示例使读者学以致用。全书共分 9 章, 在大多数章节中, 将首先对基础知识进行介绍, 然后重点讲解相关的实例。

第 1、2 章对 C# 的基础知识, 包括 C# 与 .NET 的关系、C# 语言的特点、基本语法等进行了介绍。

第 3 章对 C# 的面向对象特性进行了介绍, 读者可以从这一章中了解如何在 C# 中实现基本的面向对象特性, 如封装、继承、多态等。

第 4 章讲解了 C# 的数组和集合的相关知识。

第 5 章介绍了 C# 的字符串和正则表达式的相关知识。

第 6 章是 C# 的高级应用, 主要讲解对文件和文件夹的读、写等操作。

第 7 章是本书的重点内容之一, 介绍了数据库技术应用, 主要讲解了与 SQL Server 数据库的交互和 XML 技术的应用。

第 8 章也是本书的重点内容, 其中介绍了 Windows 窗体应用, 重点讲解了几个常用窗体控件、菜单和 GDI+ 绘图技术, 在最后几节中, 还介绍了网络通信的应用以及基于 P2P 技术的聊天程序。

第 9 章介绍了 C# 的 Web 网络应用。其中重点讲解了 ASP.NET 语法、内置对象和常用控件等。在该章最后, 还以在线投票系统为例, 深入剖析了 ASP.NET 的实际开发。

本书在每章之后都配有习题, 并且设计了上机指导, 供上机实验和实践操作使用。

本书由陈佛敏、潘春华、吕洋波共同编写完成。此外, 参与本书整理、审校和代码调试的还有陈晨、郭军军、符滔滔、梁倩、王征、王妍、刘丹等, 在此, 编者对以上人员致以诚挚的谢意! 本书中所有例题和相关代码都已调试通过, 并制作了

与本书配套的多媒体课件，供老师教学参考使用。

由于时间仓促和编者的水平有限，书中错误和不妥之处在所难免，敬请读者批评指正。

编 者

2008 年 3 月

目 录

第 1 章 C#与.NET 框架.....1

- 1.1 .NET 框架简介.....1
- 1.2 C#与.NET 的关系.....2
- 1.3 C#的特点.....2
- 1.4 Visual Studio 2005.....3
- 1.5 第一个 C#程序: Hello World.....4
 - 1.5.1 第一个 Windows 控制台应用程序.....4
 - 1.5.2 第一个 Windows 窗体应用程序.....6
 - 1.5.3 第一个 ASP.NET 应用程序.....7
- 小结.....9
- 习题.....9
- 上机指导.....9
 - 实验一 Windows 控制台应用程序.....9
 - 实验二 Windows 窗体应用程序.....9
 - 实验三 ASP.NET 应用程序.....10

第 2 章 C#语法基础.....11

- 2.1 数据类型.....11
 - 2.1.1 值类型.....11
 - 2.1.2 引用类型.....13
- 2.2 变量和常量.....14
 - 2.2.1 变量.....15
 - 2.2.2 常量.....15
- 2.3 类型转换.....16
 - 2.3.1 隐式转换.....16
 - 2.3.2 显式转换.....16
 - 2.3.3 装箱和拆箱.....17
- 2.4 运算符.....17
 - 2.4.1 算数运算符.....17
 - 2.4.2 字符串运算符.....17
 - 2.4.3 赋值运算符.....18
 - 2.4.4 逻辑运算符.....18
 - 2.4.5 位运算符.....18
 - 2.4.6 其他运算符.....19

- 2.4.7 运算符的优先级.....19
- 2.5 语句结构.....19
 - 2.5.1 条件语句.....20
 - 2.5.2 循环语句.....22
 - 2.5.3 控制循环.....23
 - 2.5.4 异常处理语句.....24
 - 2.5.5 条件编译.....25

- 小结.....25
- 习题.....26
- 上机指导.....26
 - 实验一 类型转换.....26
 - 实验二 if 条件语句.....26
 - 实验三 for 循环语句.....27

第 3 章 面向对象设计.....29

- 3.1 面向对象概述.....29
 - 3.1.1 对象的概念.....29
 - 3.1.2 面向对象的设计方法.....29
- 3.2 命名空间.....30
 - 3.2.1 命名空间的概念.....30
 - 3.2.2 命名空间的定义和引用.....30
- 3.3 类.....32
 - 3.3.1 类的概念.....32
 - 3.3.2 类的声明.....32
 - 3.3.3 类的成员和访问控制.....33
 - 3.3.4 构造函数和析构函数.....33
- 3.4 域和属性.....34
 - 3.4.1 域.....35
 - 3.4.2 属性.....35
- 3.5 抽象类.....35
 - 3.5.1 抽象类的概念.....36
 - 3.5.2 抽象类的声明.....36
 - 3.5.3 抽象方法.....37
- 3.6 接口.....38
 - 3.6.1 接口的概念.....38

3.6.2 接口的声明	38	4.4.3 元素出队	66
3.6.3 接口的实现	38	4.5 堆栈	66
3.6.4 接口与抽象类	39	4.5.1 创建堆栈	67
3.7 方法	40	4.5.2 元素入栈	67
3.7.1 方法的声明	40	4.5.3 元素出栈	67
3.7.2 参数	40	小结	68
3.7.3 静态方法	42	习题	68
3.7.4 方法的重载	43	上机指导	68
3.7.5 操作符的重载	44	实验一 使用数组	68
3.8 继承和多态	45	实验二 使用队列	68
3.8.1 继承	45	实验三 使用堆栈	69
3.8.2 多态	46		
小结	46	第5章 字符串和正则表达式	70
习题	46	5.1 字符串	70
上机指导	46	5.1.1 比较字符串	70
实验一 设计一个老师类	47	5.1.2 格式化字符串	72
实验二 使用接口求圆的面积	47	5.1.3 连接字符串	72
实验三 教师类方法的重载	48	5.1.4 分割字符串	73
第4章 数组和集合	49	5.1.5 插入字符串	73
4.1 数组	49	5.1.6 删除字符串	74
4.1.1 数组简介	49	5.1.7 遍历字符串	75
4.1.2 创建数组	50	5.1.8 复制字符串	76
4.1.3 访问数组	51	5.1.9 大小写转换	77
4.1.4 数组排序	52	5.2 StringBuilder 类	77
4.1.5 数组应用的实例	53	5.2.1 创建 StringBuilder 对象	77
4.2 数组序列	56	5.2.2 追加字符串	78
4.2.1 创建 ArrayList	57	5.2.3 插入字符串	78
4.2.2 添加元素	57	5.2.4 删除字符串	78
4.2.3 删除元素	58	5.3 正则表达式	79
4.2.4 查找元素	58	5.3.1 正则表达式简介	79
4.2.5 遍历列表	59	5.3.2 正则表达式 (Regex) 类	79
4.3 哈希表	60	5.3.3 构造正则表达式	80
4.3.1 Hashtable 类	60	5.3.4 示例: 验证 URL	82
4.3.2 构造普通哈希表	61	小结	83
4.3.3 SortedList 类	62	习题	83
4.3.4 搜索排序哈希表	63	上机指导	83
4.4 队列	65	实验一 字符串的操作	83
4.4.1 创建队列	65	实验二 使用 StringBuilder 类	84
4.4.2 元素入队	65		

第 6 章 文件操作85	7.3.2 使用 SqlCommand 获取 查询命令.....113
6.1 文件和文件夹.....85	7.3.3 使用 DataAdapter 提交 查询命令.....113
6.1.1 System.IO 类介绍.....85	7.4 管理内存数据.....114
6.1.2 文件类.....86	7.4.1 数据集简介.....114
6.1.3 文件夹类.....87	7.4.2 使用 DataTable 实现内存表.....115
6.1.4 文件信息类.....88	7.4.3 使用 DataSet 管理数据.....118
6.1.5 文件夹信息类.....89	7.4.4 使用 DataReader 获取只读数据.....120
6.1.6 文件信息类与文件夹 信息类的用法.....90	7.4.5 比较 DataSet 和 DataReader.....121
6.2 流.....90	7.5 XML 应用.....122
6.2.1 流操作类介绍.....90	7.5.1 理解 XML.....122
6.2.2 文件流类 (FileStream).....90	7.5.2 XML 相关类.....123
6.2.3 流写入类.....93	7.5.3 XML 数据的访问.....124
6.2.4 流读取类.....95	7.5.4 创建 XML 节点.....129
6.2.5 二进制流写入类.....95	7.5.5 修改 XML 节点.....130
6.2.6 二进制流读取类.....97	7.5.6 删除 XML 节点.....131
6.3 文件操作实例.....98	7.5.7 使用 DataSet 加载 XML 数据.....131
6.3.1 窗体布局.....98	小结.....133
6.3.2 代码实现.....100	习题.....133
6.3.3 实例进阶.....103	上机指导.....133
小结.....105	实验一 数据库的连接.....133
习题.....105	实验二 访问 XML 数据.....134
上机指导.....105	实验三 与 Access 数据库交互.....134
实验一 创建文件.....105	
实验二 创建文件夹.....106	
第 7 章 数据库开发技术107	第 8 章 Windows 窗体应用136
7.1 ADO.NET 简介.....107	8.1 Windows 窗体简介.....136
7.1.1 数据访问技术.....107	8.1.1 认识窗体设计器.....136
7.1.2 System.Data 命名空间.....108	8.1.2 使用窗体设计器.....138
7.2 连接数据库.....109	8.2 Windows 窗体控件.....140
7.2.1 SqlConnection 类.....109	8.2.1 按钮控件.....140
7.2.2 设置连接参数.....110	8.2.2 标签控件.....142
7.2.3 创建 SQL Server 连接.....110	8.2.3 文本框控件.....143
7.2.4 断开 SQL Server 连接.....111	8.2.4 单选按钮控件.....146
7.2.5 其他数据库连接.....111	8.2.5 复选框控件.....150
7.3 与数据库交互.....111	8.2.6 列表框控件.....153
7.3.1 使用 SqlCommand 提交增 删改命令.....111	8.2.7 可选列表框控件.....155
	8.3 菜单.....157
	8.3.1 创建菜单.....158
	8.3.2 相应菜单事件.....159

8.4 单文档和多文档应用程序	160	9.3 ASP.NET 内置对象	193
8.4.1 基于对话框的应用程序	160	9.3.1 使用 Application 对象保存数据	193
8.4.2 单文档应用程序	160	9.3.2 使用 Session 对象保存数据	193
8.4.3 多文档应用程序	163	9.3.3 访问 Server 对象	194
8.5 GDI+绘制图形	169	9.3.4 访问 Request 对象	194
8.5.1 Graphics 对象	169	9.3.5 访问 Response 对象	195
8.5.2 画笔类	169	9.4 代码绑定技术	196
8.5.3 字体类	172	9.4.1 分离显示功能和逻辑功能	196
8.5.4 位图 Bitmap 类	172	9.4.2 使用<%@ CodeFile %>绑定代码	196
8.6 网络通信	174	9.4.3 控件事件接收用户输入	198
8.6.1 套接字 (Socket)	174	9.5 使用控件	199
8.6.2 同步套接字	174	9.5.1 文本框控件	199
8.6.3 异步套接字	176	9.5.2 按钮控件	200
8.6.4 多线程技术	176	9.5.3 单选框控件	203
8.7 使用 P2P 技术的聊天程序	177	9.5.4 链接按钮控件	204
8.7.1 P2P 技术介绍	177	9.5.5 下拉框控件	205
8.7.2 创建界面	178	9.5.6 列表框控件 ListBox	207
8.7.3 监听消息	179	9.5.7 复选框控件	209
8.7.4 发送消息	179	9.5.8 图像控件	210
小结	180	9.6 Web 服务	212
习题	180	9.6.1 Web 服务简介	212
上机指导	181	9.6.2 创建 Web 服务	213
实验一 创建菜单	181	9.6.3 创建 Web 服务类	214
实验二 创建多文档应用程序	181	9.6.4 创建 Web 服务方法	215
实验三 使用 GDI+绘制图形	182	9.6.5 使用 Web 服务	216
实验四 创建一个用户登录的界面	182	9.6.6 示例: 天气预报 Web 服务	218
第 9 章 Web 网络应用	184	9.7 在线投票系统	220
9.1 ASP.NET 简介	184	9.7.1 数据库设计	220
9.1.1 什么是 ASP.NET	184	9.7.2 数据操作类的设计	220
9.1.2 IIS 管理 ASPX 页面	185	9.7.3 创建页面	228
9.2 ASP.NET 语法	187	9.7.4 投票功能	229
9.2.1 剖析 ASPX 页面	187	9.7.5 投票统计	230
9.2.2 使用<% %>嵌入代码	188	小结	232
9.2.3 使用<Script>...</Script>嵌入代码	189	习题	232
9.2.4 使用 Server 控件	191	上机指导	233
9.2.5 使用<%--注释--%>	191	实验一 使用 Session 对象保存数据	233
9.2.6 ASP.NET 2.0 新特性	192	实验二 访问 Application 对象	233
9.2.7 用<%@ Page...%>设置页面属性	192	实验三 创建 Web 服务	233
9.2.8 使用<%@ Import %>引入类库	192	实验四 使用 ASP.NET 创建一个用户 登录界面	234

第 1 章

C#与.NET 框架

自从微软公司进入.NET 时代之后,互联网领域已经发生了很大的变化。.NET 的目标是使任何人从任何地方、在任何时间、使用任何装置都能使用互联网上的服务。作为本书的开篇,首先介绍.NET 及 C#开发语言的基本知识。

1.1 .NET 框架简介

互联网的出现已经彻底改变了人类的生活方式。从静态页面到能够与用户交互的动态页面,互联网已经能够实现很强大的功能。Web 应用系统能够根据用户的要求动态处理数据,给用户提供个性化的服务。

但是现在的浏览器页面各自独立,互不相干。在互联网模式中,信息被存储在 Web 服务器内,用户的所有操作都依靠它。而无法让不同的网页互相合作,传递有意义的信息,提供更深层次的服务。

于是,微软公司设想把整个互联网变成一个操作系统,用户在互联网上开发应用程序,使用互联网上的所有应用,就好像在自己的 PC 上一样,感觉不到互联网的存在。微软公司希望“Code Once, Run Anywhere”,即写好一个程序,然后能够将其应用到任何地方,这就是.NET 的目标。整体上,.NET Framework 如图 1-1 所示。

由图 1-1 可见,.NET Framework 主要分为 4 个部分:通用语言开发环境、.NET 基础类库、.NET 开发语言和 Visual Studio.NET 集成开发环境。

1. 通用语言开发环境 (Common Language Runtime)

开发程序时,如果使用符合通用语言规范(Common Language Specification, CLS)的开发语言,那么所开发的程序将可以在任何有通用语言开发环境(Common Language Runtime)的操作系统(包括 Windows 95/98、Windows CE 及 Windows NT/2000/XP 等)下执行。

2. .NET 基础类库 (Basic Class Library)

.NET 基础类库是一套函数库,以结构严密的树状层次组织,并由命名空间(Namespace)和类(Class)组成。其功能强大,使用简单,并具有高度的可扩展性。

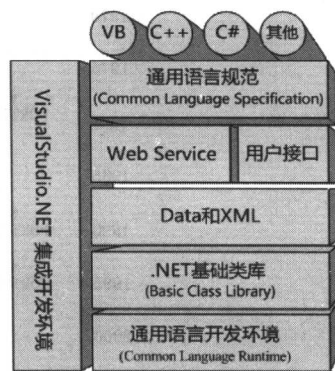


图 1-1 .NET Framework

3. .NET 开发语言

.NET 是多语言开发平台,所谓的.NET 开发语言指的是符合通用语言规范(Common Language Specification)的程序语言。目前微软公司提供 Visual Basic.NET、C#、C++以及 JScript.NET,其他厂商也提供了很多对.NET 的语言支持,其中有 APL、COBOL、Pascal、Eiffel、Haskell、ML、Oberon、Perl、Python、Scheme、Smalltalk 等。

4. Visual Studio.NET 集成开发环境

Visual Studio.NET 集成开发环境是开发.NET 应用的利器,秉承了 Microsoft IDE 一贯的易用性,功能非常强大。

1.2 C#与.NET 的关系

首先,来了解一下 C#的诞生。C 和 C++一直是最有生命力的编程语言,这两种语言提供了强大的功能、高度的灵活性以及完整的底层控制能力;缺点是开发周期较长,学习和掌握这两种语言比较困难。而许多开发效率更高的语言,如 Visual Basic,在功能方面又具有局限性。于是,在选择开发语言时,许多程序设计人员面临着两难的抉择。

针对这个问题,微软公司发布了称之为 C#(C Sharp)的编程语言。C#是为.NET 平台量身定做的开发语言,采用面向对象的思想,支持.NET 最丰富的基本类库资源。C#提供快捷的开发方式,又没有丢掉 C 和 C++强大的控制能力。C#与 C 和 C++非常相似,熟悉 C 和 C++的程序设计人员能够很快掌握 C#。C#的诞生汲取了目前所有开发语言的精华,其家谱如图 1-2 所示。

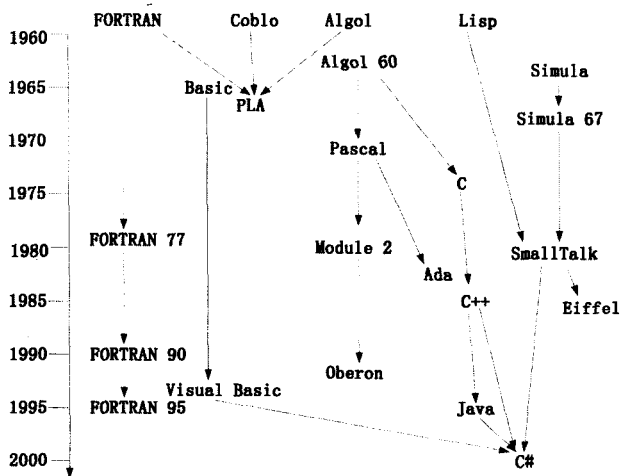


图 1-2 C#家谱

1.3 C#的特点

C#是专门为.NET 应用而开发的语言,是与.NET 框架的完美结合。在.NET 类库的支持下,C#能够全面地体现.NET Framework 的各种优点。总地来说,C#具有以下突出的优点。

1. 语法简洁

C#源自 C 和 C++。与之相比, C#最大的特色是不允许直接操作内存, 去掉了指针操作。另外, C#简化了 C++中一些冗余的语法, 如“const”和“#define”, 使语法更加简洁。

2. 彻底的面向对象设计

C#是彻底的面向对象语言, 每种类型都可以看作一个对象。C#具有面向对象语言所应有的一切特征, 包括封装、继承和多态, 并且精心设计。C#极大地提高了程序设计人员的效率, 缩短了开发周期。读者在学习和实践中, 定会感受到这一点。

3. 与 Web 应用紧密结合

C#与 Web 紧密结合, 支持绝大多数的 Web 标准, 如 HTML、XML、SOAP 等。利用简单的 C#组件, 程序设计人员能够快速开发 Web 服务, 并通过 Internet 使这些服务能被运行于任何操作系统上的应用所调用。

4. 强大的安全机制

C#具有强大的安全机制, 可以消除软件开发中许多常见错误, 并能够帮助程序设计人员尽量使用最少的代码来完成功能, 这不但减轻了程序设计人员的工作量, 同时有效地避免了错误的发生。另外, .NET 提供的垃圾回收器能够帮助程序设计人员有效地管理内存资源。

5. 完善的错误、异常处理机制

对错误的处理能力的强弱是衡量一种语言是否优秀的重要标准。在开发中, 即使最优秀的程序设计人员也会出现错误。C#提供完善的错误和异常触发机制, 使程序在交付应用时更加健壮。

6. 灵活版本处理技术

在大型工程的开发中, 升级系统的组件非常容易出错。为了处理这个问题, C#在语言本身内置了版本控制功能, 使程序设计人员更加容易地开发和维护各种商业应用。

7. 兼容性

C#遵守.NET 的通用语言规范 (Common Language Specification, CLS), 从而保证能够与其他语言开发的组件兼容。

1.4 Visual Studio 2005

本书主要使用 Visual Studio.NET (后面简称为 VS.NET) 作为开发环境, 所以需要首先介绍一下这个集成开发工具 (IDE)。VS.NET 是一个非常复杂、庞大的产品, 但其秉承了微软公司开发环境的一贯风格, 使用起来非常简单。在正确地安装配置之后, 第一次启动 VS.NET 时, 主窗口会出现一系列历史解决方案列表, 如图 1-3 所示。

VS.NET 具有以下最基本的功能。

(1) 把光标放在隐藏的窗口上, 便会自动弹出这些窗口。

(2) “工具箱”窗口显示常用的窗体控件, 通过简单的拖曳操作, 可以快速开发图形化界面工程。

(3) “解决方案资源管理器”窗口显示当前解决方案的信息 (解决方案即一个或多个工程及其配置的综合), 窗口中可以查看解决方案中工程的各种视图, 如其中的文件等。

(4) “属性”窗口显示工程内容更详细的信息, 可以对工程中的单个控件或其他对象进行配置。

例如，可以使用“属性”窗口改变 Windows 工程中一个按钮的外观、大小等。

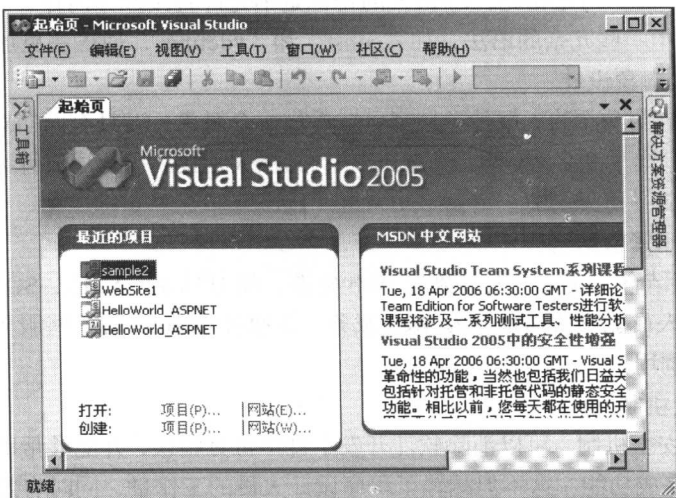


图 1-3 VS.NET 2005 启动界面

(5)“任务列表”窗口和“输出”窗口显示编译工程时的信息，以及开发环境已经完成的任務。

1.5 第一个 C#程序：Hello World

上面两节介绍了 .NET Framework 和 C# 的基本知识。本节将带领读者实现一个最简单的应用：Hello World。

1.5.1 第一个 Windows 控制台应用程序

.NET 可以实现多种应用，包括控制台程序、Windows Form 程序以及 Web 应用。首先来实现最简单的控制台程序 Hello World。实现步骤如下所述。

(1) 启动 VS.NET。

(2) 单击主窗口上“新建项目”命令，或者单击“文件”|“新建”|“项目”命令。

(3) 弹出“新建项目”对话框，如图 1-4 所示。在左侧“项目类型”中选择“Visual C#”项目，在右侧“模板”中选择“控制台应用程序”，在“名称”输入框中输入“HelloWorld_Console”，通过单击“浏览”按钮，选择工程所在的目录，单击“确定”按钮。最后，把“创建解决方案的目录”前复选框中的勾去掉。

(4) 查看“解决方案资源管理器”面板，如图 1-5 所示。右击“Program.cs”文件，在打开的菜单中单击“重命名”命令，将其改名为“HelloWorld.cs”。

(5) 查看主窗口，里面应该有 VS.NET 自动生成的代码，如下所示：

```
1. using System;
2. using System.Collections.Generic;
3. using System.Text;
4.
5. namespace HelloWorld_Console
6. {
```

```

7.     class Program
8.     {
9.         static void Main(string[] args)
10.        {
11.        }
12.    }
13. }

```

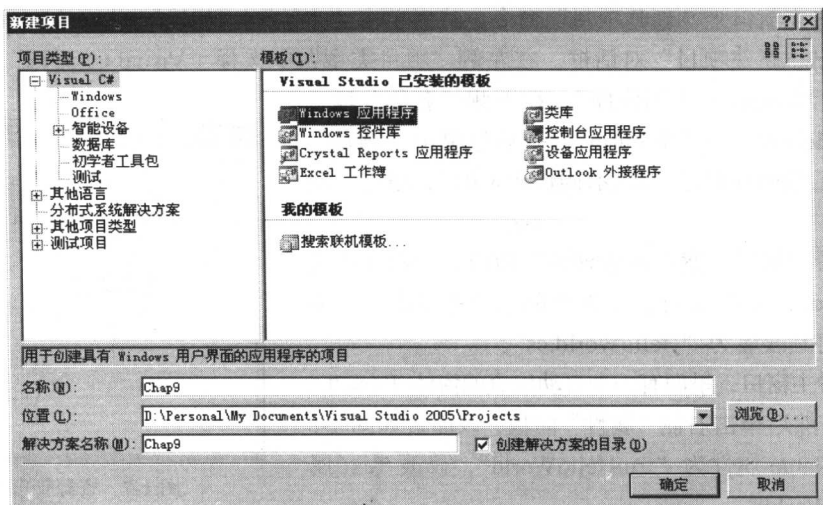


图 1-4 “新建项目”对话框

在第 7 行，将“class Program”改为“class HelloWorld”；在第 10~11 行中间，添加如下代码：
 Console.WriteLine("Hello World, Console Application.");

(6) 按快捷键“Ctrl+F5”，或者菜单操作，单击“调试”|“开始执行”命令，启动程序，运行结果如图 1-6 所示。



图 1-5 HelloWorld_Console 的资源管理器窗口

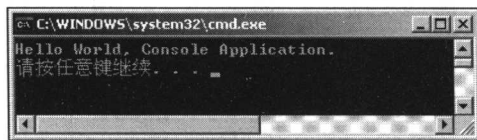


图 1-6 HelloWorld_Console 运行结果

(7) 查看工程文件。在目录“D:\示例代码\C01”下，将会发现文件夹“HelloWorld_Console”，这是 VS.NET 为本工程所建立的工程文件夹。进入后发现会包含许多文件，此处需要了解下面文件。

- HelloWorld_Console.sln: 解决方案文件，后缀“sln”为 solution 缩写，双击可以打开本工程。
- HelloWorld.cs: 工程代码文件，后缀“cs”为 C Sharp 的缩写，并非 Counter-Strike。
- 在子目录“\bin\Debug”下，可以发现可执行文件“HelloWorld_Console.exe”，双击可以执行。

(8) 如果想关闭解决方案，可单击“文件”|“关闭解决方案”命令。

至此，第一个 Windows 控制台应用就完成了。

1.5.2 第一个 Windows 窗体应用程序

本小节介绍如何在 Windows 图形化界面应用中实现 Hello World。实现步骤如下。

(1) 启动 VS.NET。

(2) 单击主窗口上“新建项目”命令，或者单击“文件”|“新建”|“项目”命令。

(3) 弹出“新建项目”对话框。在左侧“项目类型”中选择“Visual C#项目”，在右侧“模板”中选择“Windows 应用程序”，在下侧“名称”输入框中输入“HelloWorld_WinForm”，并通过单击“浏览”按钮，选择工程所在目录“D:\示例代码\C01”；单击“确定”按钮。

(4) 查看“解决方案资源管理器”窗口，如图 1-7 所示。右击“Form1.cs”文件，在弹出的菜单中单击“重命名”命令，将其改名为“HelloWorld.cs”。

(5) 查看主窗口，里面有一个自动生成的窗体 Form1。单击该窗体，然后单击右侧“属性”面板，该面板如图 1-8 所示。修改 Name 属性为“frmHelloWorld”，修改 Text 属性为“Hello World!”。

(6) 单击主窗口左侧的“工具箱”面板，出现一些 Windows 控件，如图 1-9 所示。



图 1-7 资源管理器窗口

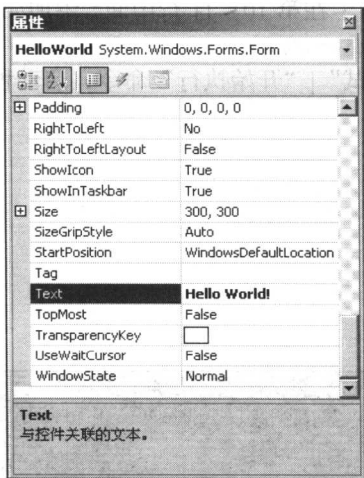


图 1-8 Form1 的“属性”窗口

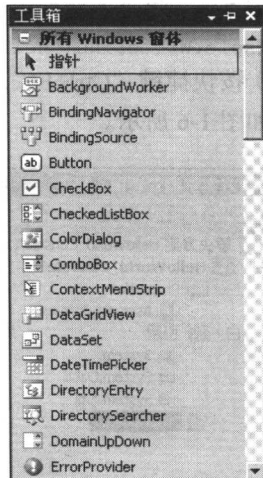


图 1-9 Windows Form 工具箱

(7) 双击“Label”控件，或者单击后按住左键将其拖曳至主窗口的窗体中，并修改其属性。

- Name: lblDisplay。
- Text: 空。
- BackColor: Window。

(8) 双击“Button”控件，或者单击后按住左键将其拖曳至主窗口的窗体中，并修改其属性。

- Name: btnShow。
- Text: “显示”。

最后效果如图 1-10 所示。

(9) 双击“显示”按钮，将进入代码窗口（通过主窗口上侧的标签可以在代码窗口和窗体窗口间进行切换），可以看到 VS.NET 已经自动生成了很多代码，在此不必关心。

(10) 进入代码窗口后光标自动位于方法“btnShow_Click()”内部（即单击“显示”按钮会触发这个方法），在光标处添加如下代码：

```
this.lblDisplay.Text="Hello World, WinForm Application.";
```

(11) 按快捷键“Ctrl+F5”，或者单击“调试”|“开始执行”命令，启动程序后，单击“显示”按钮，在标签中就会出现如图 1-11 所示的结果。

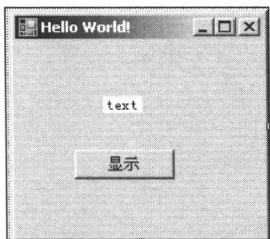


图 1-10 添加 Label 和 Button 后的窗体

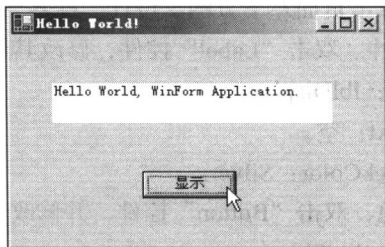


图 1-11 HelloWorld_WinForm 运行结果

(12) 查看在目录“D:\示例代码\C01”下的工程文件，将会发现文件夹“HelloWorld_WinForm”。至此，第一个 Windows Form 应用就完成了。

1.5.3 第一个 ASP.NET 应用程序

前面两个应用都是 Windows 应用，下面介绍使用 C#实现的 Web 应用，即 ASP.NET 程序。本例实现步骤如下。

(1) 启动 Visual Studio.NET 2005。

(2) 单击主窗口上的“新建项目”链接，弹出“新建网站”对话框。

(3) 选择“ASP.NET 网站”项，在下侧“位置”下拉框中选择“文件系统”，然后输入“D:\示例代码\HelloWorld_ASPNET”。在“语言”下拉框中选择“Visual C#”，如图 1-12 所示。

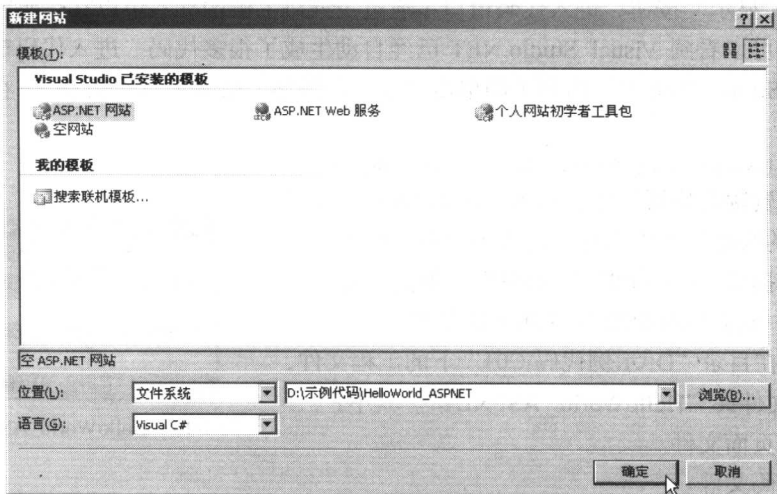


图 1-12 新建 ASP.NET 网站