

高等学校计算机科学与技术教材

- ① 原理与技术的完美结合
- ② 教学与科研的最新成果
- ③ 语言精炼，实例丰富
- ④ 可操作性强，实用性突出

# Visual C# .NET 程序设计 (修订本)

□ 杨晓光 编著

□ 李兰友 主审

清华大学出版社

● 北京交通大学出版社

高等学校计算机科学与技术教材

# Visual C# .NET 程序设计

(修订本)

杨晓光 编著  
李兰友 主审

清华大学出版社  
北京交通大学出版社  
·北京·

## 内 容 简 介

本书以言简意赅的语言、丰富而实用的大量实例代码全面阐述了 Visual C# .NET 程序设计技术，特别是 .NET 2.0 的新技术。主要内容包括：Visual C# .NET 集成开发环境的使用，C# 语言基础，C# 面向对象特性，简单界面的 Windows 程序设计，复杂界面的 Windows 程序设计，图形、图像处理，数据库应用，Web 程序设计，综合实例讲解等。通过对本书的学习和实践，可对 C# 语言有一个全面而深刻的理解，能够在 .NET 环境中利用 C# 设计 Windows 应用程序、Web 应用程序和基于数据库的应用程序。

本书可作为高等院校计算机与信息技术及相关专业的教科书，也适用于各类工程技术人员和程序设计人员参考使用。

版权所有，翻印必究。举报电话：010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术，用户可通过在图案表面涂抹清水，图案消失，水干后图案复现；或将表面膜揭下，放在白纸上用彩笔涂抹，图案在白纸上再现的方法识别真伪。

### 图书在版编目(CIP)数据

Visual C# .NET 程序设计/杨晓光编著. —修订本. —北京：清华大学出版社；北京交通大学出版社，2006.9

(高等学校计算机科学与技术教材)

ISBN 7-81082-148-2

I. V… II. 杨… 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 (2006) 第 063670 号

责任编辑：谭文芳

出版者：清华大学出版社 邮编：100084 电话：010-62776969 <http://www.tup.com.cn>

北京交通大学出版社 邮编：100044 电话：010-51686414 <http://press.bjtu.edu.cn>

印刷者：北京东光印刷厂

发行者：新华书店总店北京发行所

开本：185×260 印张：24.5 字数：627 千字

版次：2004 年 9 月第 1 版 2006 年 9 月第 1 次修订 2006 年 9 月第 4 次印刷

书号：ISBN 7-81082-148-2/TP·104

印数：14 001~19 000 册 定价：34.00 元

---

本书如有质量问题，请向北京交通大学出版社质监组反映。对您的意见和批评，我们表示欢迎和感谢。

投诉电话：010-51686043, 51686008；传真：010-62225406；E-mail: [press@center.bjtu.edu.cn](mailto:press@center.bjtu.edu.cn)。

# 前 言

自从美国微软公司推出 .NET 以来,.NET 技术在业界得到极大认同,并在短时间内得到迅速推广,越来越多的人转移到这个平台上来。而C#在 .NET 中,扮演着举足轻重的角色,.NET 框架主要是由它实现。因此,C#在 .NET 中具有先天的优势,学习C#无疑是迅速进入 .NET 的一个捷径。

C#既具有 C++ 语言功能强大、灵活,又具有 VB 的简单易学、高效的优点,利用它可以方便快捷地开发基于文字处理、图形图像、数据库、多媒体、网络的 Windows 应用程序和 Web 应用程序,它对于迅速崛起的 XML 和 Web Service 也提供了全方位的支持。因此,越来越多的人选择它作为程序设计语言,越来越多的高校和培训机构开设了这门课程。为了适应这方面的教学需要,编者结合多年的教学实践和程序设计经验编写了这本教材。

本书力求以简明扼要的语言阐述各个知识点,同时辅以大量代码和实例,以帮助读者领会和应用这些知识点。本书每章的末尾均配以综合实例,以帮助读者综合应用本章知识点。全书末尾提供了四个典型实例,以帮助读者融会贯通本书内容。

本书第 1 章介绍 Visual Studio .NET 集成开发环境;第 2 章和第 3 章从语言角度,讲述C#语言的基础知识和C#面向对象程序设计的基本思想及方法,同时介绍C#2.0 的语言新特性;第 4 章介绍一些程序设计中经常用到的对象;第 5 章和第 6 章讲解 Windows 窗体基本知识、常用窗体控件的使用,以及 Windows 应用程序界面的设计方法和实例,并穿插介绍 .NET2.0 新控件;第 7 章介绍对话框及文件操作;第 8 章和第 9 章讲解图形设计技术和图像处理技术及应用实例;第 10 章介绍利用 ADO .NET 技术开发数据库应用;第 11 章介绍利用 ASP .NET 技术开发 Web 应用;第 12 章给出四个典型实际应用程序设计的例子,它们可作为综合练习之用。

本书所提供的程序实例简明易懂,全部程序均在 Visual Studio 2005 和 Windows XP 环境下调试通过。有需要源代码的读者可在 <http://press.bjtu.edu.cn> 网站下载。

本书配有习题集和实验指导,将于近期出版,敬请关注。

本书可作为高等院校计算机专业和信息安全类专业本科教材,亦适合工程技术人员参考。

本书由杨晓光编著,李兰友教授主审。其他编写人员有马延宏、傅岚岚、杨晓君、郑志荣、郭文平等。由于作者水平有限,加之 .NET 技术复杂而又不断推陈出新,书中纰漏和考虑不周之处在所难免,敬请专家和读者批评指正。

编 者  
2006 年 4 月

# 目 录

第 1 章 Visual C# .NET 集成开发环境 .....	1
1.1 创建应用程序 .....	1
1.1.1 创建控制台应用程序 .....	1
1.1.2 创建 Windows 应用程序 .....	5
1.2 Visual Studio 主要窗口及用法 .....	12
1.2.1 起始页窗口 .....	12
1.2.2 解决方案资源管理器对话框 .....	13
1.2.3 工具箱对话框 .....	14
1.2.4 属性对话框 .....	14
1.2.5 窗体设计器窗口 .....	15
1.2.6 类视图对话框 .....	17
1.2.7 代码编辑器窗口 .....	18
1.2.8 服务器资源管理器对话框 .....	20
1.3 Visual C# .NET 的菜单栏和工具条 .....	20
1.3.1 文件菜单 .....	20
1.3.2 编辑菜单 .....	21
1.3.3 视图菜单 .....	21
1.3.4 重构菜单 .....	22
1.3.5 项目菜单 .....	22
1.3.6 生成菜单 .....	22
1.3.7 调试菜单 .....	23
小结 .....	23
习题 .....	23
第 2 章 C#语言基础 .....	24
2.1 数据类型 .....	24
2.1.1 简单类型 .....	24
2.1.2 结构类型 .....	28
2.1.3 枚举类型 .....	29
2.1.4 数组类型 .....	29
2.1.5 类类型 .....	33
2.1.6 类型转换 .....	33
2.1.7 变量 .....	35
2.2 表达式 .....	36
2.2.1 一元运算符 .....	36
2.2.2 算术运算符 .....	37
2.2.3 位运算符 .....	37
2.2.4 关系和类型测试运算符 .....	38
2.2.5 条件逻辑运算符 .....	40
2.2.6 条件运算符 .....	40

2.2.7 赋值运算符 .....	40
2.2.8 其他特殊运算符 .....	40
2.2.9 运算符的优先级 .....	42
2.3 程序控制语句 .....	43
2.3.1 选择语句 .....	43
2.3.2 循环语句 .....	46
2.3.3 跳转语句 .....	48
2.3.4 异常处理 .....	50
2.3.5 编译和调试 .....	51
2.4 综合实例 1:解线性方程组 .....	56
小结 .....	59
习题 .....	59
<b>第 3 章 C#面向对象程序设计</b> .....	<b>61</b>
3.1 类 .....	61
3.1.1 类声明 .....	61
3.1.2 创建类实例 .....	62
3.1.3 类成员 .....	62
3.1.4 构造函数和析构函数 .....	63
3.1.5 方法 .....	64
3.1.6 字段与属性 .....	71
3.1.7 继承 .....	73
3.1.8 多态 .....	75
3.1.9 抽象类 .....	77
3.1.10 密封类 .....	78
3.1.11 静态类 .....	78
3.1.12 分部类 .....	80
3.2 接口 .....	81
3.2.1 接口声明 .....	81
3.2.2 接口实现 .....	82
3.3 泛型 .....	82
3.3.1 泛型类 .....	83
3.3.2 泛型方法 .....	85
3.3.3 约束 .....	85
3.4 委托与事件 .....	88
3.4.1 委托 .....	88
3.4.2 匿名方法 .....	90
3.4.3 事件 .....	92
3.5 综合实例 2:时钟 .....	93
小结 .....	98
习题 .....	98

<b>第 4 章 常用对象</b> .....	100
4.1 Convert 类.....	100
4.2 String 类 .....	101
4.3 DateTime 结构 .....	103
4.4 Color 结构.....	105
4.5 Point, Size 和 Rectangle 结构 .....	105
4.6 Math 类.....	107
4.7 Random 类 .....	107
4.8 ArryList 类 .....	108
4.9 Hashtable 类.....	109
4.10 综合实例 3: 数制转换 .....	110
小结.....	112
习题.....	112
<b>第 5 章 Windows 窗体和常用控件</b> .....	114
5.1 Windows 窗体 .....	114
5.1.1 窗体属性 .....	114
5.1.2 窗体方法 .....	118
5.1.3 窗体事件 .....	119
5.2 文本控件 .....	122
5.2.1 Label 控件 .....	122
5.2.2 TextBox 控件.....	124
5.3 按钮控件 .....	127
5.4 列表选择控件 .....	130
5.4.1 ListBox 控件 .....	130
5.4.2 ComboBox 控件 .....	133
5.5 选择控件 .....	136
5.5.1 RadioButton 控件和 GroupBox 控件 .....	136
5.5.2 CheckBox 控件 .....	139
5.6 图形显示控件 .....	140
5.7 定时控件 .....	142
5.8 鼠标和键盘事件 .....	143
5.8.1 鼠标事件 .....	143
5.8.2 键盘事件 .....	145
5.9 综合实例 4: 复杂计算器 .....	147
小结.....	152
习题.....	152
<b>第 6 章 Windows 应用程序界面设计</b> .....	153
6.1 单窗体界面设计 .....	153
6.1.1 界面设计原则 .....	153
6.1.2 单窗体界面设计 .....	154

6.2	菜单界面设计	157
6.2.1	菜单编辑器	157
6.2.2	MenuStrip 控件和 ToolStripMenuItem 控件	159
6.2.3	菜单界面设计	160
6.3	工具栏界面设计	161
6.3.1	ToolStrip 控件和 ToolStripButton 控件	161
6.3.2	ImageList 控件	162
6.3.3	工具栏界面设计	163
6.4	状态栏界面设计	165
6.4.1	StatusStrip 控件和 ToolStripStatusLabel 控件	165
6.4.2	状态栏界面设计	166
6.5	MDI 界面设计	167
6.5.1	建立 MDI 主窗体	168
6.5.2	建立 MDI 子窗体	168
6.5.3	主窗体程序设计	170
6.5.4	子窗体程序设计	170
6.5.5	合并菜单	171
6.6	多窗体界面设计	172
6.6.1	建立多窗体	172
6.6.2	窗体间关联	173
6.6.3	多窗体界面设计	174
6.7	综合实例 5:精灵	179
	小结	185
	习题	185
<b>第 7 章</b>	<b>对话框</b>	<b>186</b>
7.1	自定义对话框	186
7.1.1	模态对话框	186
7.1.2	非模态对话框	187
7.2	通用对话框	189
7.2.1	消息框	189
7.2.2	文件打开和保存对话框	190
7.2.3	颜色对话框	193
7.2.4	字体对话框	193
7.2.5	打印对话框	193
7.3	综合实例 6:文本编辑器	194
7.3.1	RichTextBox 控件	194
7.3.2	文本编辑器界面设计	195
7.3.3	文本编辑器程序设计	196
	小结	199
	习题	199



<b>第 8 章 C#图形程序设计</b> .....	201
8.1 GDI+ 绘图基础 .....	201
8.1.1 图形设备接口 .....	201
8.1.2 创建 Graphics 对象 .....	201
8.1.3 创建画笔对象 .....	202
8.1.4 创建画刷 .....	203
8.1.5 字体与文本 .....	206
8.2 基本图形绘制 .....	207
8.3 实用图形程序设计 .....	214
8.4 综合实例 7:贝济埃曲线 .....	219
小结 .....	222
习题 .....	222
<b>第 9 章 图像处理</b> .....	223
9.1 Visual C#图像处理基础 .....	223
9.1.1 图像文件类型 .....	223
9.1.2 Bitmap 对象 .....	223
9.2 图像的输入和保存 .....	224
9.2.1 图像的输入 .....	224
9.2.2 图像的保存 .....	226
9.3 图像拷贝和粘贴 .....	228
9.3.1 使用剪贴板拷贝和粘贴图像 .....	228
9.3.2 使用 AxPictureClip 控件拷贝和粘贴图像 .....	230
9.4 彩色图像处理 .....	231
9.4.1 图像的分辨率 .....	231
9.4.2 彩色图像变换灰度图像 .....	233
9.4.3 灰度图像处理 .....	234
9.5 动画 .....	237
9.6 综合实例 8:彩图处理 .....	239
小结 .....	249
习题 .....	249
<b>第 10 章 数据库应用</b> .....	251
10.1 概述 .....	251
10.2 ADO .NET 对象 .....	252
10.2.1 Connection 对象 .....	252
10.2.2 Command 对象 .....	254
10.2.3 DataReader 对象 .....	259
10.2.4 DataAdapter 对象 .....	261
10.2.5 DataSet 对象 .....	262
10.2.6 数据控件及数据绑定控件 .....	264
10.3 综合实例 9:ADO .NET 应用 .....	273
10.3.1 实例简介 .....	274
10.3.2 实例分析设计 .....	274

10.3.3	创建项目	274
10.3.4	创建数据库	274
10.3.5	主控程序	277
10.3.6	雇员管理子系统	280
10.3.7	顾客管理子系统	284
10.3.8	其他子系统	286
小结		286
习题		286
<b>第 11 章</b>	<b>Web 应用</b>	<b>287</b>
11.1	ASP .NET 基础	287
11.1.1	ASP .NET 工作原理	287
11.1.2	运行前的准备工作	288
11.1.3	创建 ASP .NET Web 应用程序	289
11.1.4	ASP .NET 语法和内置对象	291
11.1.5	ASP .NET 事件处理	298
11.2	ASP .NET 服务器控件	300
11.2.1	使用 ASP .NET 服务器控件	300
11.2.2	HTML 服务器控件	301
11.2.3	Web 服务器控件	305
11.2.4	验证控件	322
11.3	Web 服务	324
11.3.1	创建 Web 服务	325
11.3.2	使用 Web 服务	327
11.4	XML	328
11.4.1	XML 架构	328
11.4.2	创建 XML	330
11.4.3	使用 XML	330
11.5	综合实例 10:留言本	332
小结		337
习题		337
<b>第 12 章</b>	<b>应用程序设计实践</b>	<b>338</b>
12.1	电子邮箱	338
12.1.1	POP3 协议简介	338
12.1.2	电子邮箱程序	338
12.2	Web 浏览器	354
12.2.1	WebBrowser 控件简介	354
12.2.2	Web 浏览器	354
12.3	统计图表	366
12.4	媒体播放器	372
小结		381
习题		381

# 第 1 章 Visual C# .NET 集成开发环境

## 本章要点:

---

Visual C# .NET 是微软公司系列可视化开发工具 Visual Studio .NET 中的产品,是创建应用程序最简便、最快捷的开发工具之一。本章主要介绍如何创建 Visual C# .NET 应用程序及集成开发环境的基础知识和基本操作。重点掌握:

- 创建 Visual C# .NET 应用程序
  - Visual C# .NET 的窗口对象的意义
  - Visual C# .NET 主菜单的使用方法
- 

## 1.1 创建应用程序

Visual C# .NET 语言功能强大,使用它可以开发控制台应用程序、Windows 应用程序和 Web 应用程序。本章主要讨论如何开发控制台应用程序和 Windows 应用程序,而 Web 应用程序的开发留待第 11 章讨论。

### 1.1.1 创建控制台应用程序

控制台应用程序是 C# 能够创建的最基本类型之一,它在控制台窗口进行输出。下面通过创建一个在控制台窗口输出“欢迎进入 C# 多彩世界”的简单例子讲述创建控制台应用程序的基本步骤。

#### (1) 启动 Visual Studio 2005

从开始菜单启动 Visual Studio 2005,屏幕出现如图 1-1 所示的起始页。

#### (2) 创建新项目

在【文件】菜单中,选择【新建】中的【项目】选项,则显示如图 1-2 所示的【新建项目】对话框。

在【新建项目】对话框中,在【项目类型】栏中选择【Visual C#】;在【模板】栏中选择【控制台应用程序】;在【名称】文本框中填写应用程序的名称(本例为“WelcomeConsole”);在【位置】文本框中输入项目保存位置“e:\ Visual C# .NET \ ch01”(此位置只是一个示例,用户可以自行决定保存位置,而不必与此相同);在【解决方案】下拉列表框中选择【创建新解决方案】;在【解决方案名称】文本框中输入解决方案名称“WelcomeConsole”。选中【创建解决方案的目录】复选框,这样可以为项目创建一个新文件夹,用来保存项目的所有文件。

**注意:**在【名称】文本框写入应用程序的项目名,可以是中文或英文字符串。

单击【确定】按钮,则显示控制台应用程序代码编辑窗口,如图 1-3 所示。在代码编辑窗口内显示系统自动生成的相应代码。

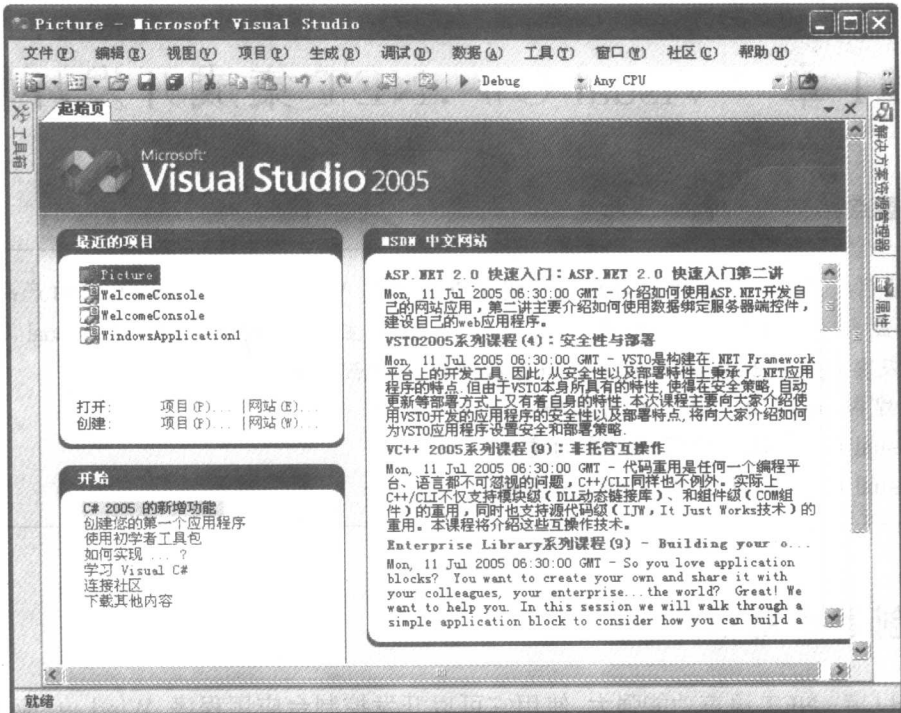


图 1-1 Visual Studio 2005 起始页

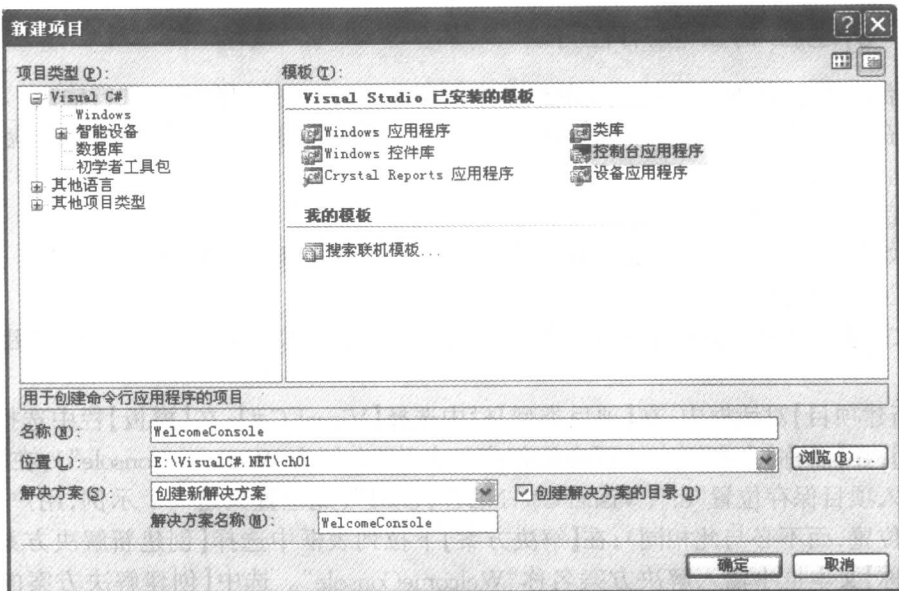


图 1-2 【新建项目】对话框

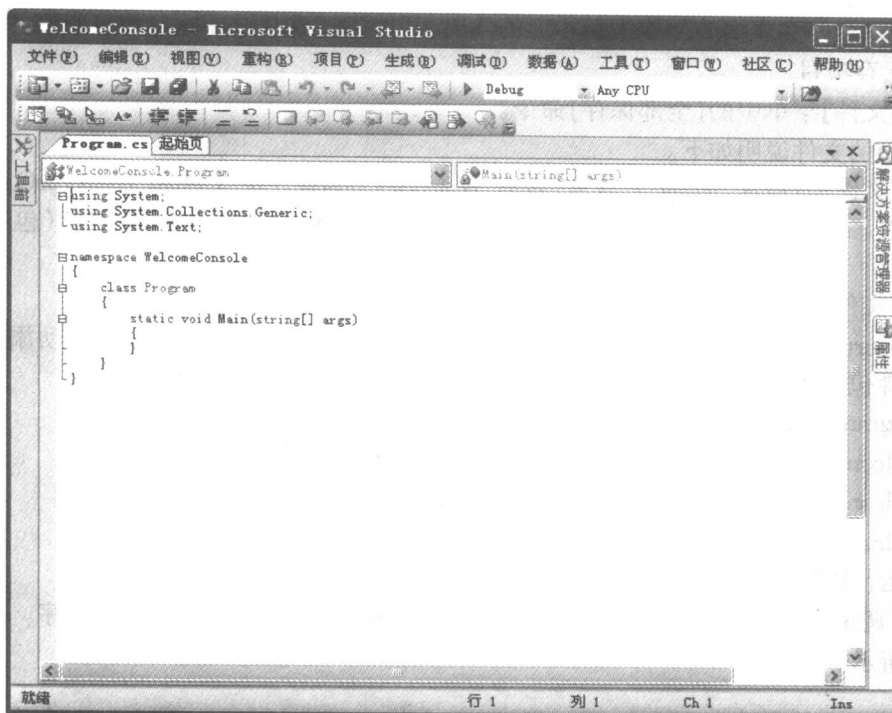


图 1-3 控制台应用程序代码编辑窗口

### (3) 编写代码

以下为程序的全部代码,其中,每行代码前的数字是为了便于分析代码后加上去的,并不包含在实际代码中。

```
1. //Sample 1.1:WelcomeConsole
2. //一个C#程序的简单示例
3.
4. using System;
5. using System.Collections.Generic;
6. using System.Text;
7.
8. namespace WelcomeConsole
9. {
10.     class Program
11.     {
12.         static void Main(string[] args)
13.         {
14.             Console.WriteLine("欢迎进入C#多彩世界!");
15.             Console.Read();
16.         }
17.     }
18. }
```

其中,以黑体字表示的部分为新添加的代码,其他为集成开发环境自动生成的代码。

#### (4) 保存项目

选择【文件】菜单中的【全部保存】命令。

项目中的文件说明如下。

☞ bin 文件夹:包含 debug 子目录,含有 WelcomeConsole.exe 文件和 WelcomeConsole.pdb 文件。.exe 文件为生成的可执行文件,.pdb 文件包含完整的调试信息(包括函数原型)。

☞ obj 文件夹:包含 debug 子目录,含有编译过程中生成的中间代码。

☞ Properties 文件夹:包含 AssemblyInfo.cs 文件,它是在创建项目过程中自动添加的。此文件包含程序集属性的设置。

☞ program.cs 文件:应用程序文件,包含应用程序代码。

☞ WelcomeConsole.csproj:项目文件。

☞ WelcomeConsole.csproj.user:用户配置文件。

☞ WelcomeConsole.csproj.sln:解决方案文件。

#### (5) 运行程序

选择【调试】菜单中的【开始执行(不调试)】命令或按 Ctrl + F5 键来运行程序。运行结果如图 1-4 所示。

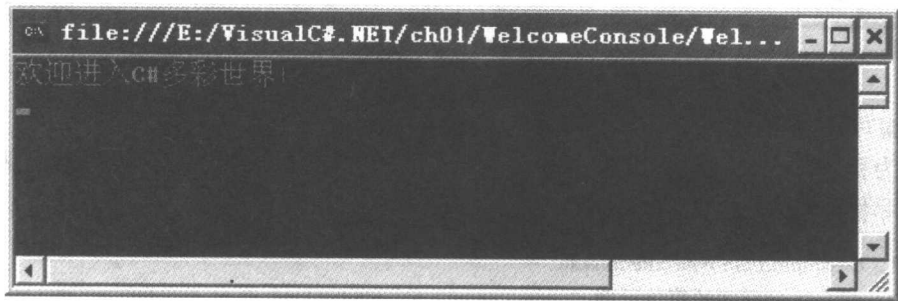


图 1-4 运行结果

#### (6) 结束程序

单击控制台窗口的【关闭】按钮,即可结束程序的运行。

#### 【程序分析】

程序的第 1 行以“//”开头,表示该行为注释。在代码中加入注释是为了提高代码的可读性。它可以帮助其他人阅读和理解程序。C#中的注释分为两种,以“//”开头的注释为单行注释,注释在该行的末尾结束;另一种为多行注释,注释的内容放在“/\*”和“\*/”中间。

程序的第 3 行为空行。使用空行可以增加程序的可读性,因此建议读者编写代码时插入必要的空行。

程序的第 4 行使用 using 指令导入命名空间。.NET 框架提供了非常丰富的类,这些类是按照命名空间的层次结构组织的。通过命名空间组织各个类可以帮助避免二义性,减少命名冲突,以及将相关的类组组在一起。例如,如果需要处理数据,可引入 System.Data 命名空间。

程序的第 8 行使用 namespace 关键字定义命名空间。当使用 Visual Studio .NET 创建应

用程序时,代码将放置于与应用程序名称相同的命名空间中。通过定义命名空间可以更好地组织代码和避免命名冲突。

程序第 10~17 行使用 `class` 关键字定义一个类 `Program`,它完成应用程序的主要任务。C#程序一般由用户自定义的类和 .NET 框架中提供的类所组成,并且要求程序至少要包含一个类。

程序的第 12 行定义了一个 `Main` 方法。它是控制台应用程序的入口点,程序中必须包含 `Main` 方法,否则程序将不会被执行。请注意 `Main` 方法的名字以大写的“M”开头,不要写成小写的“m”,因为 C#语言是区分大小写的,写成小写的“m”编译时会报错。

程序的第 14 行使用 `Console.WriteLine` 方法向控制台窗口输出字符串“欢迎进入 C#多彩世界!”。`WriteLine` 方法是 .NET 框架的 `Console` 类的一个静态方法,它有许多重载函数,最常用的有以下两种形式:

```
Console.WriteLine("欢迎进入C#多彩世界!");
```

用于输出常量字符串。

```
Console.WriteLine("x = {0}", x);
```

用于输出变量 `x` 的值。其中,“{0}”为占位符,输出时它会被变量 `x` 的值所代替。

程序的第 15 行使用 `Console.Read` 方法使程序停顿下来,等待用户按 `Enter` 键。主要是为了避免程序输出一闪而过,以至于看不清楚输出的内容。

## 1.1.2 创建 Windows 应用程序

Windows 应用程序也是 C# 能够创建的最基本类型之一,它使用各种图形界面元素显示一个 Windows 窗口。下面通过创建一个在窗口中显示一幅图片的简单例子讲述创建 Windows 应用程序的基本步骤。

### (1) 启动 Visual Studio 2005

从开始菜单启动 Visual Studio 2005。

### (2) 创建新项目

在【文件】菜单中,选择【新建】中的【项目】选项,则显示如图 1-2 所示的【新建项目】对话框。

在【新建项目】对话框中,在【项目类型】栏中选择【Visual C#】;在【模板】栏中选择【Windows 应用程序】;在【名称】文本框中填写应用程序的名称(本例为“Picture”);在【位置】文本框中输入项目保存位置(本例为“e:\Visual C# .NET\ch01”);在【解决方案】下拉列表框中,选择【创建新解决方案】;在【解决方案名称】文本框中输入解决方案名称(本例为 Picture)。选中“创建解决方案的目录”复选框。

单击【确定】按钮,则显示窗体设计器窗口,如图 1-5 所示。在窗体设计器窗口中,显示一个默认窗体。

### (3) 在窗体上添加一个图片框

从工具箱的【公共控件】选项卡中把 `PictureBox` 控件拖到窗体上,调整图片框位置和大小,如图 1-6 所示。

### (4) 设置图片框的属性

单击图片框控件,则在【属性】对话框中显示图片框控件的属性,如图 1-7 所示。

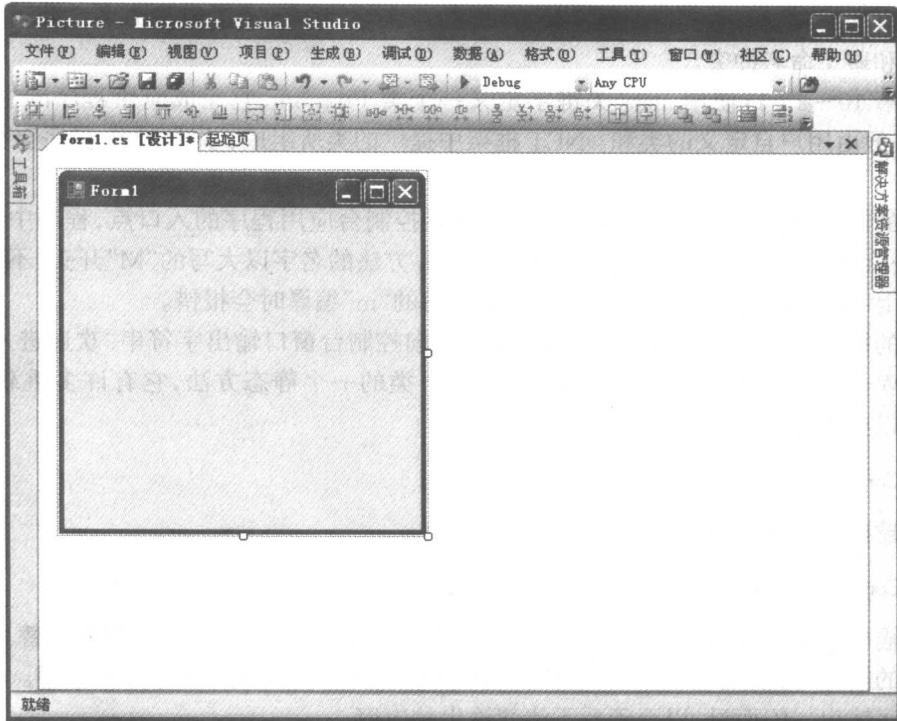


图 1-5 窗体设计器窗口

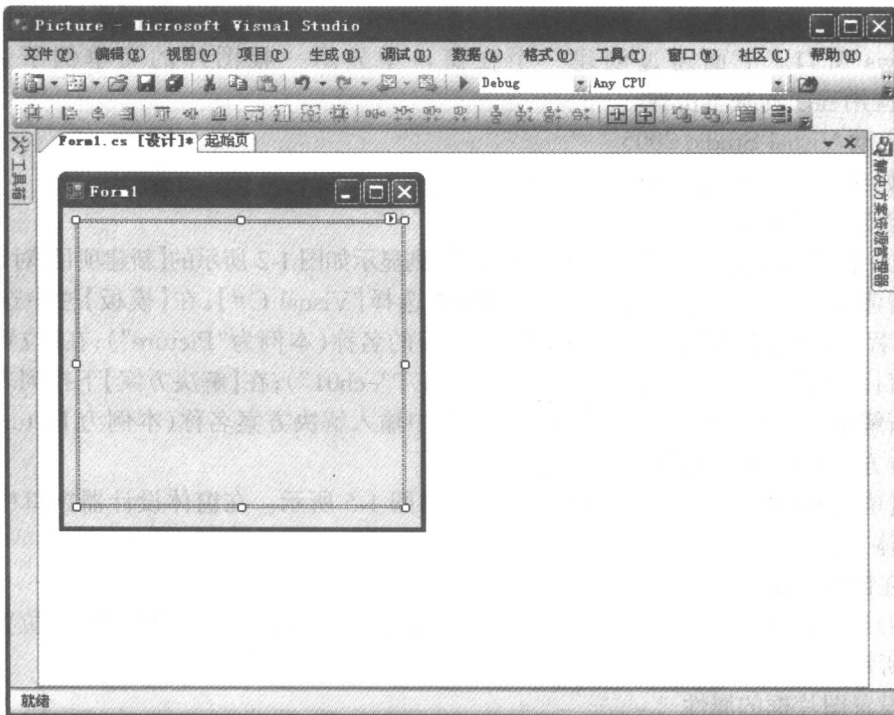


图 1-6 带有图片框的窗体设计器窗口





图 1-7 显示 PictureBox 属性的【属性】对话框

找到 Image 属性,单击右侧的【…】按钮,则显示【选择资源】对话框,如图 1-8 所示。

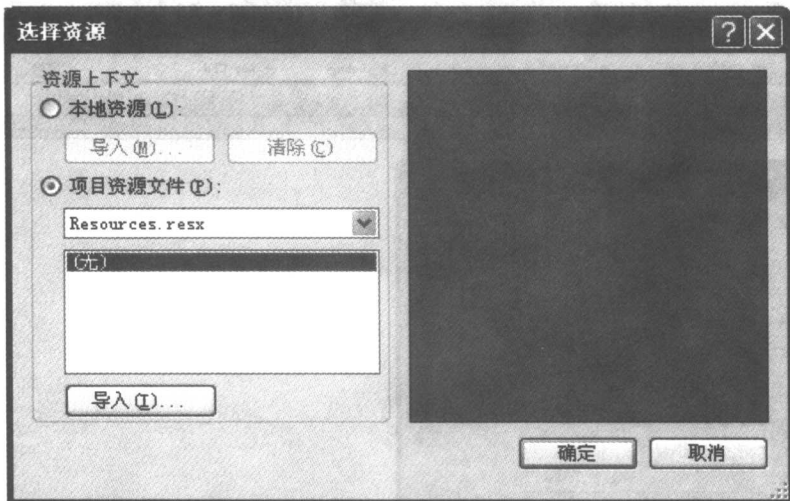


图 1-8 【选择资源】对话框

单击【导入】按钮,弹出【打开】对话框,如图 1-9 所示。选择一个图片文件,单击【打开】按钮,返回到【资源选择】对话框。单击【确定】按钮,关闭【资源选择】对话框。

#### (5) 在窗体上添加一个按钮控件

从工具箱把 Button 控件拖到窗体上,并调整好位置和大小,如图 1-10 所示。

#### (6) 设置按钮的属性

选中按钮控件,在【属性】对话框中找到 Text 属性,给该属性赋值为“退出”。