



国家示范（骨干）高职院校
重点建设专业优质核心课程系列教材

郑广成 沈蕴梅 周玲余 董 连 编著

Python 程序设计 项目化教程



中国水利水电出版社
www.waterpub.com.cn

国家示范（骨干）高职院校重点建设专业

优质核心课程系列教材

C#程序设计项目化教程

郑广成 沈蕴梅 周玲余 董 连 编著



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

C#语言作为 VS.NET 的框架语言,倍受专业爱好者和从业人员的青睐。本书由浅入深、循序渐进,采取典型的项目载体,采取课内外项目并行、工作过程项目化的模式,系统地介绍和训练了 C#控制台程序、流程控制程序、Windows 窗体程序、面向对象的设计与系统实现、XML 技术和文件操作等,重点讲述了基于数据库的 Windows 应用程序开发、面向对象的继承、多态、集合和构造函数等技术在应用系统开发中的应用。为了便于读者全面掌握程序设计技术和规范,深刻体会编程的乐趣,最后给出一个综合性的实战项目,全面讲述了以数据库为基础的应用系统开发全过程。

本书的单元内容和项目,完全对照高职高专学生实际的能力要求,在作者多年的实践教学过程中总结提炼而成,采取新的“工作过程项目化”的教学流程进行内容重组,实现了课内主讲一个项目、课外训练一个项目、示例依托案例项目的形式,通过工作场景导入、技术与知识准备、回到工作场景、工作训练营的流程实现各单元内容的编写。

本书提供配套教学课件和各单元源代码程序供读者参考,可以从中国水利水电出版社和万水书苑的网站上免费下载,网址为: <http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

图书在版编目(CIP)数据

C#程序设计项目化教程 / 郑广成等编著. — 北京 : 中国水利水电出版社, 2012.10

国家示范(骨干)高职院校重点建设专业优质核心课程系列教材

ISBN 978-7-5170-0299-4

I. ①C… II. ①郑… III. ①C语言—程序设计—高等教育—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2012)第253262号

策划编辑: 石永峰

责任编辑: 宋俊娥

封面设计: 李 佳

书 名	国家示范(骨干)高职院校重点建设专业优质核心课程系列教材 C#程序设计项目化教程
作 者	郑广成 沈蕴梅 周玲余 董 连 编著
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658 (发行部)、82562819 (万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市铭浩彩色印装有限公司
规 格	184mm×260mm 16开本 14.25印张 350千字
版 次	2012年10月第1版 2012年10月第1次印刷
印 数	0001—3000册
定 价	26.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

VS.NET 已成为面向对象的程序开发主流平台, 它可以开发 Windows 应用程序、Web 应用程序、嵌入式软件应用程序、游戏程序等各种应用系统软件, 受到相关行业内专业人士和学习者的青睐。

本书主要基于岗位技能、软件流程和规范, 采取“工程过程项目化”的模式编写, 是教学团队结合“产教融合、产学并行”的教学改革和实践总结出来的教学模式和教学内容的展现, 团队经过筛选和提炼后, 确定典型项目作为教学内容载体, 更适合以应用能力为本位的高职高专、应用型本科的教学、训练的要求。

本书通过引入项目场景提出工作问题, 通过示例项目训练技能、解决工作问题并讲解相应的技术和方法, 然后回到工作场景完成场景项目, 再通过工作训练营项目进一步训练, 采用课内外项目并行推进的教学形式, 达到巩固所学知识和举一反三的训练效果。本书的内容打破了传统的学科章节和硬项目化编写形式, 采取教学(工作)单元的形式进行内容组编。全书分成 11 个单元, 最后一个单元通过综合实训训练学生技能, 进一步提高学生的应用实践能力, 体现了“做中学、学中产”的实训教学思想。本书主要内容如下:

- 单元一 熟悉 C#开发环境
- 单元二 储备 C#知识和技术
- 单元三 设计流程控制程序
- 单元四 使用方法、数组与 String 类
- 单元五 开发 Windows 应用程序
- 单元六 设计面向对象程序
- 单元七 使用集合开发程序
- 单元八 使用继承和多态开发程序
- 单元九 TreeView 进行 XML 数据显示和操作
- 单元十 文件操作
- 单元十一 综合项目实训

本书由郑广成、沈蕴梅、周玲余、董连编写, 郑广成负责统稿并担任主编, 参加部分章节编写工作的还有朱翠苗、许戈, 在此一并表示感谢。本书根据技术模块设置单元, 根据典型项目设计内容载体, 通过课内外两个项目并行推进来提高学生的应用能力和创新能力, 具有实战性、可操作性、新颖新、通俗性和项目过程化的特点, 更加激发学生学习兴趣和主动性。

由于时间仓促, 再加上编者水平有限, 书中难免有错误和疏漏之处, 敬请广大读者批评指正。

编 者
2012 年 5 月

目 录

前言

单元一 熟悉 C#开发环境	1
1.1 工作场景导入	1
1.2 技术与知识准备	2
1.2.1 认识 C#	2
1.2.2 第一个 C#控制台应用程序	2
1.2.3 第一个 Windows 窗体应用程序	6
1.3 回到工作场景	10
1.4 工作训练营	11
1.4.1 项目训练	11
1.4.2 工作实践常见问题解析	13
小结	13
单元二 储备 C#知识和技术	14
2.1 工作场景导入	14
2.2 技术与知识准备	15
2.2.1 基本概念	15
2.2.2 运算符与表达式	17
2.2.3 Console 类	21
2.3 回到工作场景	22
2.4 工作训练营	22
2.4.1 项目训练	22
2.4.2 工作实践常见问题解析	24
小结	25
单元三 设计流程控制程序	26
3.1 工作场景导入	26
3.2 技术与知识准备	27
3.2.1 条件判断	27
3.2.2 while 型循环	36
3.2.3 do-while 型循环	38
3.2.4 for 型循环	39
3.2.5 其他控制语句	42
3.3 回到工作场景	43
3.4 工作训练营	45
3.4.1 项目训练	45

3.4.2 工作实践常见问题解析	46
小结	46
单元四 使用方法、数组与 String 类	48
4.1 工作场景导入	49
4.2 技术与知识准备	49
4.2.1 方法	49
4.2.2 数组	52
4.2.3 foreach 语句	53
4.2.4 C#中的 String 类	53
4.2.5 冒泡排序	54
4.3 回到工作场景	54
4.4 工作训练营	56
4.4.1 项目训练	56
4.4.2 工作实践常见问题解析	58
小结	58
单元五 开发 Windows 应用程序	59
5.1 工作场景导入	59
5.2 技术与知识准备	60
5.2.1 ADO.NET	60
5.2.2 NET Framework 数据提供程序	60
5.2.3 DataSet	69
5.3 回到工作场景	73
5.4 工作训练营	76
5.4.1 项目训练	76
5.4.2 工作实践常见问题解析	81
小结	81
单元六 设计面向对象程序	82
6.1 工作场景导入	82
6.2 技术与知识准备	83
6.2.1 类、对象和封装	83
6.2.2 字段和属性	83
6.2.3 构造函数	87
6.2.4 方法重载	90

6.2.5 静态成员	92	9.2.2 创建 XML 文档	141
6.3 回到工作场景	94	9.2.3 解析 XML 文件	142
6.4 工作训练营	96	9.2.4 用 TreeView 显示数据	143
6.4.1 项目训练	96	9.2.5 用 ListBox 显示 XML 文件	145
6.4.2 工作实践常见问题解析	97	9.3 回到工作场景	147
小结	98	9.4 工作训练营	152
单元七 使用集合开发程序	99	9.4.1 项目训练	152
7.1 工作场景导入	99	9.4.2 工作实践常见问题解析	155
7.2 技术与知识准备	100	小结	155
7.2.1 集合	100	单元十 文件操作	156
7.2.2 泛型	105	10.1 工作场景导入	156
7.2.3 泛型集合 Dictionary<k,v>	108	10.2 技术与知识准备	157
7.3 回到工作场景	110	10.2.1 文件操作的类	157
7.4 工作训练营	114	10.2.2 序列化和反序列化	166
7.4.1 项目训练	114	10.3 回到工作场景 1	169
7.4.2 工作实践常见问题解析	117	10.4 回到工作场景 2	172
小结	117	10.5 工作实训营	174
单元八 使用继承和多态开发程序	119	10.5.1 训练实例	174
8.1 工作场景导入	119	10.5.2 工作实践常见问题解析	176
8.2 技术与知识准备	120	小结	176
8.2.1 继承	120	单元十一 综合项目实训	177
8.2.2 多态	126	11.1 综合项目实训说明	177
8.2.3 面向对象的三大特性	130	11.1.1 实训目的	177
8.3 回到工作场景	130	11.1.2 实训对象	177
8.4 工作训练营	134	11.1.3 实训项目	177
8.4.1 项目训练	134	11.1.4 实训要求	177
8.4.2 工作实践常见问题解析	137	11.2 宿舍管理信息系统	178
小结	137	11.2.1 系统功能	178
单元九 TreeView 进行 XML 数据显示和操作	138	11.2.2 数据库设计	179
9.1 工作场景导入	138	11.2.3 系统详细设计与实现	182
9.2 技术与知识准备	139	实训 阶段项目测试	214
9.2.1 XML 文件概述	139	参考文献	221

单元一

熟悉 C# 开发环境



单元要点

- Visual Studio 2010 开发环境
- 控制台应用程序和 Windows 应用程序的内涵和区别
- 创建和运行控制台应用程序的方法和步骤
- 创建和运行 Windows 应用程序的方法和步骤



技能目标

- 能使用控制台编写简单的应用程序
- 能独立设计 Windows 窗体应用程序



项目载体

- 工作场景项目：欢迎员工加入
- 工作训练营项目：图片放大和缩小

1.1 工作场景导入

【工作场景】

小张才从某高校毕业，应聘到某软件公司的 .NET 开发团队，公司主管要小王编写一个“欢迎小张加入 .NET 团队！”的程序，要求编写控制台程序和 Windows 应用程序两种形式，效果如图 1.1 和图 1.2 所示。



图 1.1 控制台程序效果

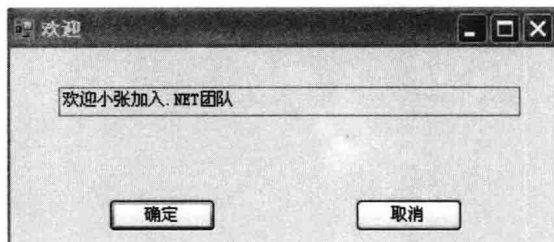


图 1.2 Windows 应用程序效果

【引导问题】

- (1) 如何用 Microsoft Visual Studio 2010 开发 C#控制台应用程序?
- (2) 如何用 Console 类进行控制台输入和输出?
- (3) 如何用 Microsoft Visual Studio 2010 开发 C#窗体应用程序?

1.2 技术与知识准备

1.2.1 认识 C#

C# (C sharp) 语言是从 C 和 C++ 演变而来的,但在诸如命名空间、类、方法和异常处理等特定领域,与 C 和 C++ 还是存在着巨大的差异;C# 是一种简单、现代、类型安全和面向对象的编程语言;Microsoft 公司设计 C# 是为了建立运行在 .NET 平台上的、范围广泛的企业级应用程序;用 Visual C# 编写的代码被编译为托管代码,这意味着它将受益于公共语言运行库的服务,这些服务包括语言互操作性、垃圾回收、增强的安全性以及改进的版本支持。在 Visual 2010 中, C# 完全得到项目模板、设计器、属性页、代码助理、对象模型和其他开发环境功能的支持。

C# 已成为在 .NET 平台上进行开发的首选语言,微软甚至用 C# 编写了 .NET 框架的大部分代码, C# 功能强大且具有灵活性,以前的 Visual Basic 具有简单、易用等特性, Visual Studio 2010 将它的特性与 C# 融于一体,大大方便了程序员快速进行网络开发,同时保持开发者所需要的强大性和灵活性。

1.2.2 第一个 C# 控制台应用程序

【示例 1.1】创建一个简单的 C# 控制台程序。

工作步骤如下:

【步骤 1】启动 Visual Studio 2010 开发工具。

【步骤 2】依次选择“文件”→“新建”→“项目”菜单,打开“新建项目”对话框,如图 1.3 所示。可以看到,左边的是项目类型,右边的是已安装的模板,包括“Windows 窗体应用程序”、“类库”、“控制台应用程序”等模板,它们指定了要创建的应用程序的类型。

【步骤 3】在左侧选择“Visual C#”,右侧选择“控制台应用程序”,名称框中输入“Welcome”,并选择项目的存放位置,如图 1.4 所示。



图 1.3 “新建项目”对话框



图 1.4 输入名称并选择存放位置

【步骤 4】确认“为解决方案创建目录”复选框已被选中，然后单击“确定”按钮。出现如图 1.5 所示的项目设计界面。

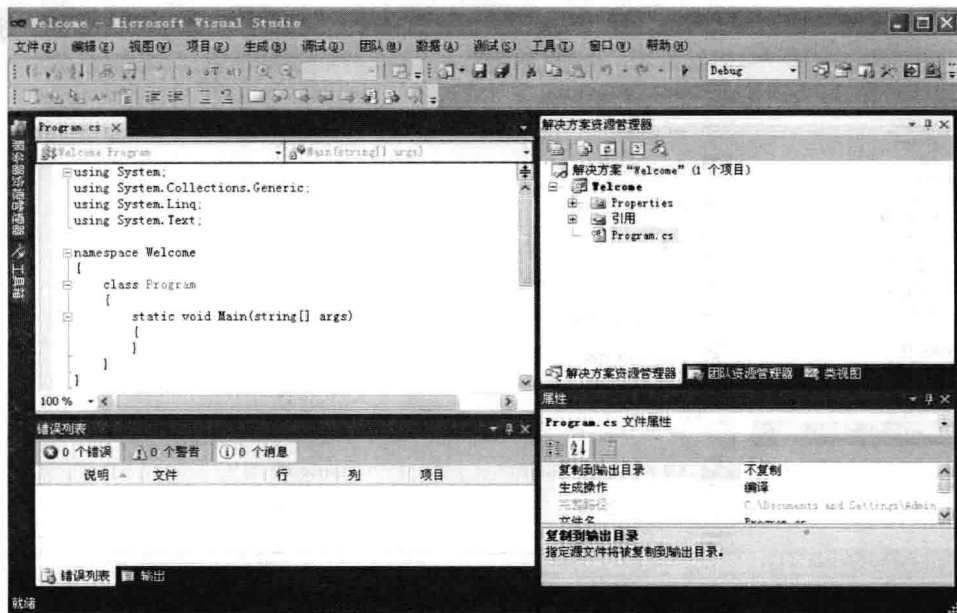


图 1.5 Welcome 项目设计界面

注意：在开始编写代码之前，先来解释一些名词。

(1) namespace 关键字。namespace（命名空间）是 C# 中组织代码的方式，这样可以把紧密相关的一些代码放在同一个命名空间中，大大提高管理和使用的效率。在这段代码中，VS2010 自动以项目的名称 Welcome 作为命名空间的名称。

(2) using 关键字。在 C# 中，使用 using 关键字来引用其他命名空间。在生成这段代码模板时，VS 就已经自动添加了 4 条 using 语句。

(3) class 关键字。C# 是一种面向对象的语言，使用 class 关键字表示类，编写的任何代码都应该包含在一个类中，类要包含在一个命名空间中。在程序模板生成时，VS 自动起了一个类名 Program。如果不喜欢，可以改掉它。

(4) Main 方法。C# 中的 Main() 方法是程序的大门，应用程序从这里开始运行。要注意的是，C# 中 Main() 方法的首字母必须大写，Main() 方法的返回值可以 void 或 int，Main() 方法中的命令行参数是可以没有的。

(5) 解决方案“Welcome”。它是最顶级的解决方案文件，每个应用程序都有一个类似的文件。根据该项目的创建路径，找到 Welcome 项目文件夹，就可以看到该解决方案文件，即 Welcome.sln。每个解决方案文件都包含对一个或者多个项目文件的引用。“Welcome”是 C# 的项目文件，其名称为 Welcome.csproj。每个项目文件都引用一个或者多个包含项目源代码的文件。

(6) Properties。查看 Welcome 项目文件夹可以知道，Properties 是其中的一个文件夹，包含一个名为 AssemblyInfo.cs 的文件，它是一个特殊文件，可以用它在一个属性中添加如“作者”、“日期”等属性。

(7) 引用。该文件包含对程序可用的已编译代码的引用。

(8) Program.cs。从代码窗口上方的选项卡中，也可以找到 Program.cs，很明显，它是一个 C# 源代码文件，用户编写的代码都包含在这个文件中，同时 VS2010 自动创建的一些源代码也保存在其中。

【步骤 5】编写代码。

前面介绍的 Program.cs 文件中, 包含一个名为 Program 的类, 该类中有一个名为 Main 的方法。Main 是一个特殊的类, 它指定了 C# 程序的入口, 即任何的 C# 程序都是从 Main 方法开始执行的。

注意: C# 语言区分字母的大小写, 因此, 在拼写 Main 时, 要注意字母的大小写。

下面为 Welcome 项目添加以下代码。

```
namespace Welcome //空间名
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("welcome! "); //控制台 Console 类方法
        }
    }
}
```

上面的代码中, Console 是一个类, WriteLine 是 Console 类的一个方法, 用于在控制台中输出文本。

【步骤 6】生成并运行控制台应用程序。

编写好程序代码后, 接下来应当生成控制台应用程序, 即编译代码并生成一个可执行程序, 具体方法是: 选择“生成”→“生成 welcome”菜单项, 生成的过程中会在代码编辑器的下方出现一个输出窗口, 如图 1.6 所示。

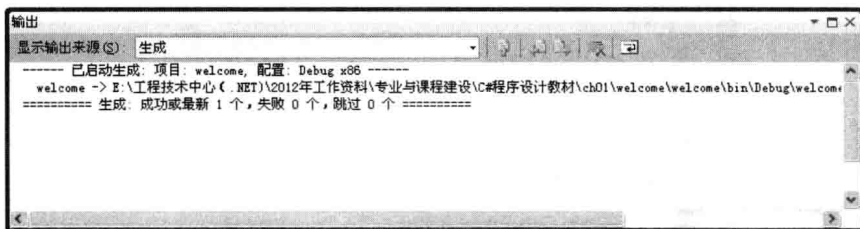


图 1.6 输出窗口

在输出窗口中, 提示程序编译完成, 并显示生成过程中发生的错误和警告。本程序非常简单, 只包含一行代码, 在输入正确的情况下, 不会有任何错误和警告。

生成成功后, 可以运行程序以查看结果。按 F5 键或单击  按钮, 即弹出一个命令窗口显示程序的运行结果, 如图 1.7 所示。



图 1.7 运行结果

1.2.3 第一个 Windows 窗体应用程序

窗体应用程序即 Windows 应用程序，它允许以图形的方式进行人机交互。下面用一个简单的实例介绍创建 Windows 应用程序的基本步骤和方法。

【示例 1.2】单击窗体 Form1 上的“确定”按钮后，在窗体上方的文本框中显示“欢迎进入 Visual C# 2010 编程世界”的字样；单击“退出”按钮，则关闭窗口，退出应用程序。

该实例的开发过程虽然简单，却体现了使用 Visual C# 2010 开发 Windows 应用程序的基本流程。主要包括 4 个工作过程。

【步骤 1】界面设计。

(1) 启动 Microsoft Visual Studio 2010，进入 Visual C# 2010 开发界面。

(2) 选择“文件”→“新建”→“项目”，弹出如图 1.8 所示的对话框，可以看到，左边是项目类型，右边是已安装的模板，包括“Windows 窗体应用程序”、“类库”、“控制台应用程序”等模板，它们指定了要创建的应用程序的类型。

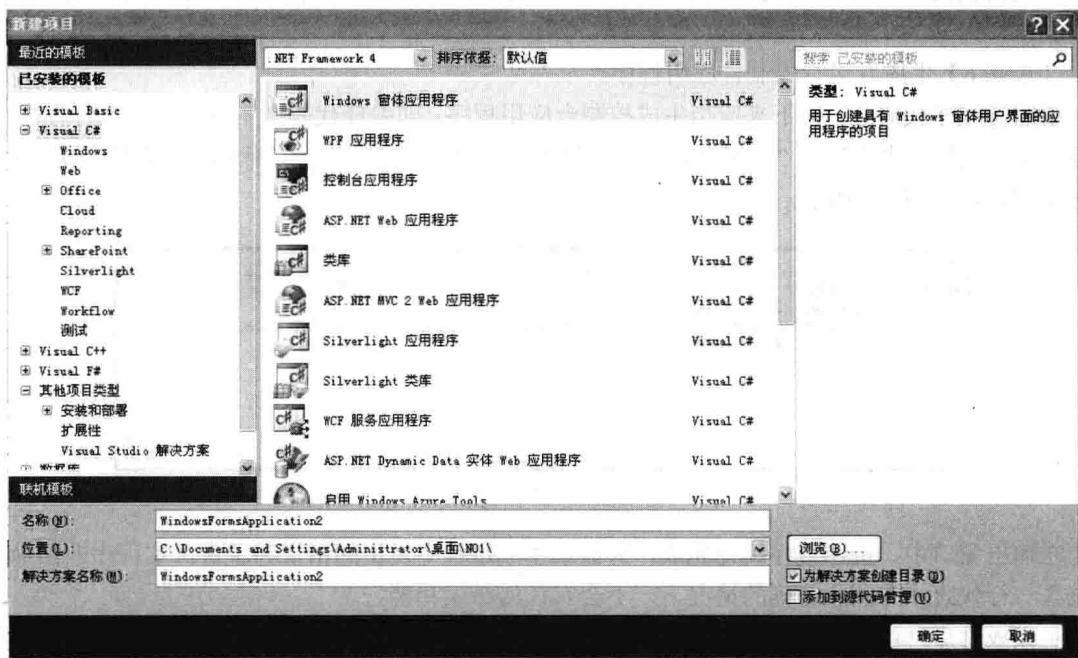


图 1.8 “新建项目”对话框

(3) 在左侧选择“Visual C#”，右侧选择“Windows 窗体应用程序”，名称框中输入“welcome2”，并选择项目的存放位置，如图 1.9 所示。

(4) 确认“为解决方案创建目录”复选框已被选中，然后单击“确定”按钮。出现如图 1.10 所示的窗体。

(5) 调整窗体到合适的大小，长宽比为 2:1，然后展开工具箱中的“所有 Windows”选项卡，找到并双击控件  TextBox，为窗体添加一个文本框控件，这时的窗体 Form1 如图 1.11 所示。

注意：添加控件也可以先在工具箱中选择相应的控件，然后在窗体中适当的位置使用鼠标拖动来实现。

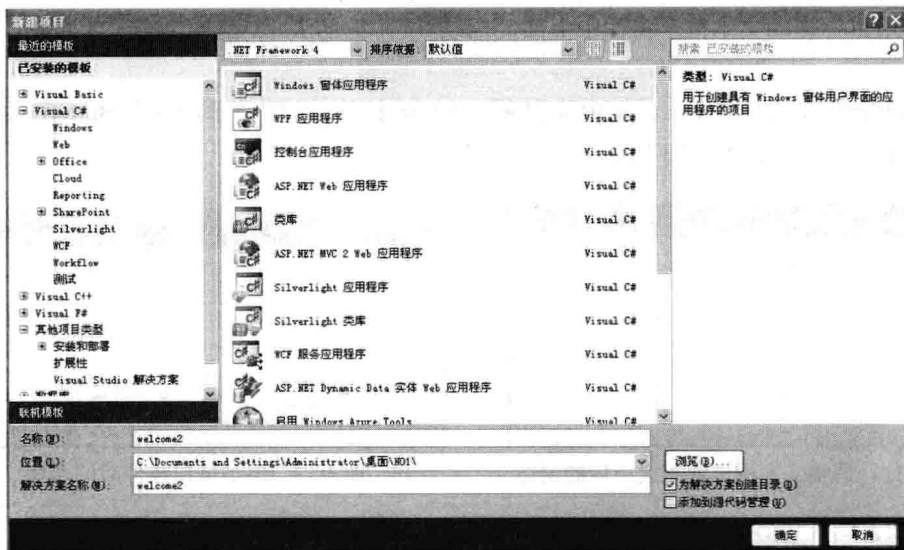


图 1.9 输入名称并选择存放位置

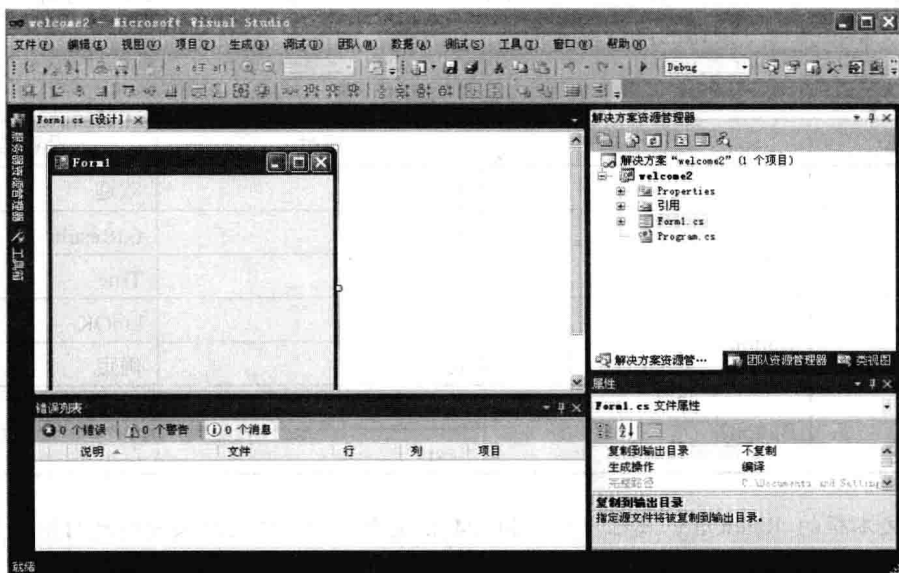


图 1.10 welcome2 项目

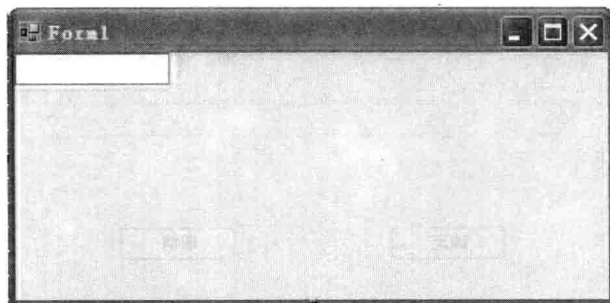


图 1.11 调整大小并添加文本框后的窗体

(6) 使用鼠标拖动文本框至窗体中上部, 并调整文本框的大小, 调整后的窗体 Form1 如图 1.12 所示。

(7) 按照同样的方法, 在工具箱中找到  Button 控件, 为窗体添加两个 Button (命令按钮), 并调整其大小和位置, 如图 1.13 所示。

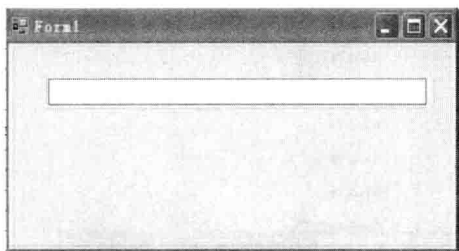


图 1.12 调整文本框位置和大小后的窗体

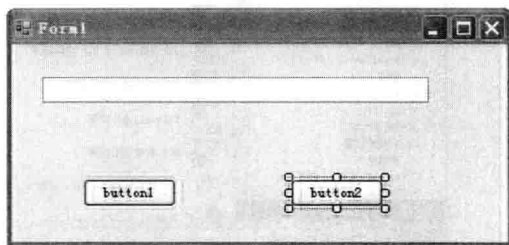


图 1.13 添加按钮控件后的窗体

【步骤 2】属性设置。

控件添加完毕后, 接下来对窗体及窗体上各控件的属性进行设置。如果属性窗口是隐藏的, 则用鼠标右击需要设置属性的对象, 在弹出的下拉菜单中选择“属性”菜单命令, 打开属性窗口。

窗体、文本框和命令按钮的属性设置如表 1.1 所示。

表 1.1 控件的属性设置

控件类型	控件名称	属性	设置结果
Form	Form1	Text	欢迎
TextBox	TextBox1	Name	txtResult
		ReadOnly	True
Button	Button1	Name	btnOK
		Text	确定
	Button2	Name	btnCancel
		Text	取消

注意: 文本框的 ReadOnly 属性用于控制文本框是否只读 (即是否接受用户的输入), 在本程序中, 文本框仅用于显示文本, 所以应当将其设为 True。

设置好窗体及各控件的属性后用户界面如图 1.14 所示。

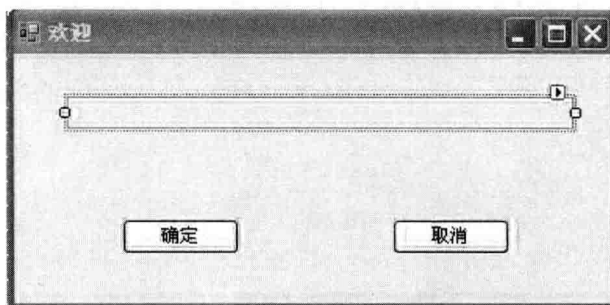


图 1.14 设置属性后的界面

可以看到,文本框变成灰色,表明该文本框是只读的,在程序运行的过程中,用户不能改变其中的文本内容。

注意: 常用控件的命名,一般遵循如表 1.2 所示的规则。

表 1.2 常用控件命名规则

控件	命名前缀	控件	命名前缀
 Button	btn	 Label	lbl
 ComboBox	cbo	 GroupBox	grp
 RadioButton	rdo	 Timer	tmr

【步骤 3】编写代码。

接下来在 btnOK 的 Click 事件中加入以下代码:

```
private void btnOK_Click(object sender, EventArgs e)
{
    Txtresult.text = "欢迎进入 Visual C# 2010 编程世界!";
}
```

注意: 该代码的含义是在文本框控件 txtResult 中显示“欢迎进入 Visual C# 2010 编程世界!”字样。

切换到用户界面窗口,再双击“取消”按钮,按照同样的方法在 btnCancel 对象的 Click 事件中加入如下代码:

```
private void btnCancel_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

到此,整个程序的代码全部添加完成,代码中的两个代码段(方法)都是用了鼠标的 Click(单击)事件。

在窗体和代码都设计好后,应当保存文件,以防止调试或运行程序时发生死机等意外而造成数据丢失,保存文件可以选择“文件”→“保存”或“全部保存”命令,或者单击工具栏上的“保存”或“全部保存”按钮来实现,另外编译程序时也会自动保存所有项目文件,编译成功后不需要再使用菜单命令来保存文件。

【步骤 4】运行程序。

到此,应用程序设计的前期工作已经完成,下一步是调试和运行程序,运行程序的方法是:单击“调试”→“启动”命令,或者单击工具栏中的按钮,还可以直接按 F5 键。运行界面效果如图 1.15 所示。

单击“确定”按钮,则会在窗体上方的文本框中显示“欢迎进入 Visual C# 2010 编程世界!”字样,如图 1.16 所示。

最后,单击“取消”按钮,则窗体关闭,并结束应用程序的运行。

注意: 如果程序不能正常运行,编译器会给出相应的提示,可以对程序运行修改,直到程序能正常运行为止。

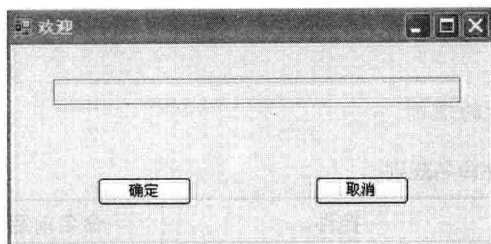


图 1.15 运行界面

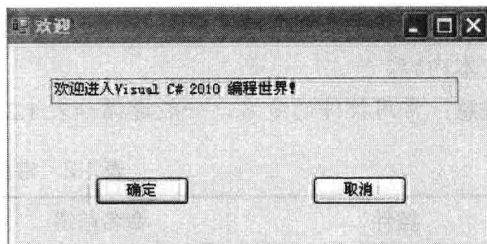


图 1.16 单击“确定”按钮的运行结果

1.3 回到工作场景

通过对以上内容的学习，了解了控制台应用程序和 Windows 窗体应用程序的创建步骤，现在回到 1.1 节工作场景的任务中。

参照 1.2 节中的技术和方法，完成“欢迎小张加入.NET 团队”程序项目。

1. 使用控制台程序实现“欢迎小张加入.NET 团队”项目

【步骤 1】新建控制台应用程序 ch01-1（参考 1.2 节中的方法和步骤，此处略）。

【步骤 2】编写代码。

```
static void Main(string[] args)
{
    Console.WriteLine("欢迎小张加入.NET 团队!");
}
```

运行效果如图 1.17 所示。

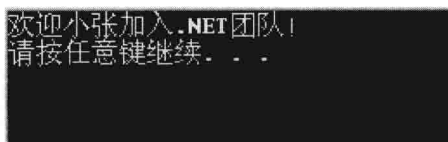


图 1.17 控制台运行结果

2. 使用 Windows 窗体应用程序实现“欢迎小张加入.NET 团队”项目

【步骤 1】新建 Windows 应用程序 ch01-2，设计界面中控件的属性设置如表 1.3 所示。

表 1.3 界面控件的属性设置

控件类型	控件名称	属性	设置结果
Form	Form1	Text	欢迎
TextBox	TextBox1	Name	txtResult
		ReadOnly	True
Button	Button1	Name	btnShow
		Text	确定
	Button2	Name	btnCancel
		Text	取消

【步骤 2】编写代码。

```
private void btnShow_Click(object sender, EventArgs e)
{
    txtWelcome.Text = "欢迎小张加入.NET 团队";    //信息显示在文本框中
}
```

运行效果如图 1.2 所示。

1.4 工作训练营

1.4.1 项目训练

1. 训练内容

小刘到 Z 公司 .NET 团队面试，公司给他一个“图片放大和缩小”的小型项目进行入职测试。要求设计一个 Windows 窗体程序，对界面的一张图像进行放大与缩小操作，具体效果如图 1.18 所示，要求在 20 分钟内完成。

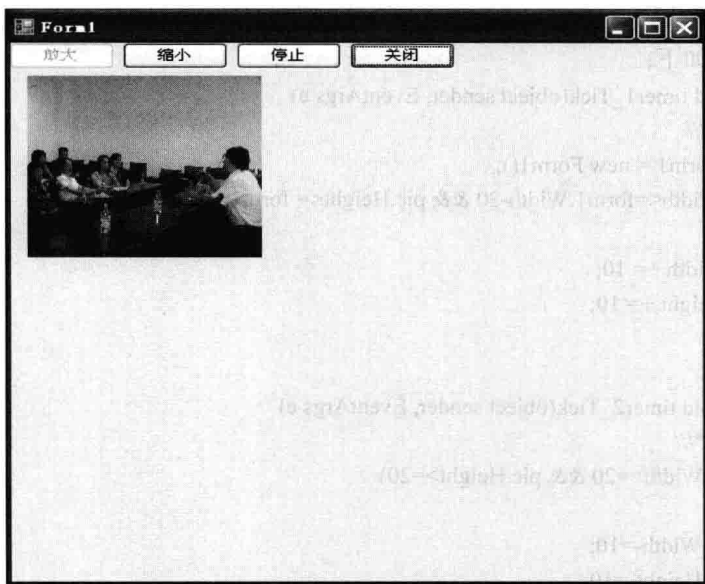


图 1.18 图片放大与缩小

2. 训练目的

- (1) 进一步训练和巩固学生对 VS2010 创建 Windows 应用程序的步骤和过程的理解；
- (2) 使学生对 C# 编程有一个比较深刻的印象和认识。

3. 训练过程

【步骤 1】 创建一个 Windows 窗体应用程序。

请参考 1.2 节中的方法和步骤。

【步骤 2】 设计窗体界面。

设计如图 1.18 所示的界面，界面中控件的属性设置如表 1.4 所示。