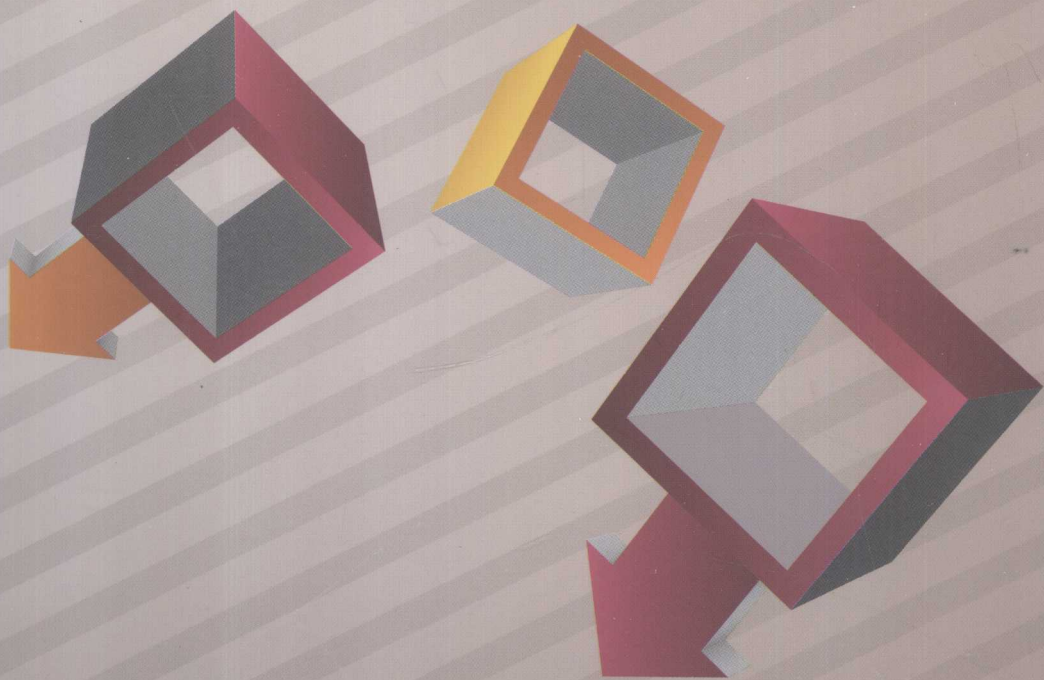


软件工程师培养丛书

WinForm技术应用

武汉厚溥教育科技有限公司 编著



“理论→总结→上机→习题”四阶段教学模式

- ★ 理论结合实践，注重动手能力培养
- ★ 任务驱动讲解，有效激发学习兴趣
- ★ 典型项目案例，扎实培养专业素质
- ★ 教学做一体化，极大提高教学效率



清华大学出版社

计算机工程与网络应用

WinForm技术应用

清华大学出版社 清华大学计算机系 编著



本书以WinForm技术为基础，结合.NET Framework 4.0，详细介绍了WinForm技术的应用。本书共分10章，主要内容包括：WinForm技术概述、WinForm窗体设计、WinForm窗体事件、WinForm窗体数据、WinForm窗体动画、WinForm窗体特效、WinForm窗体布局、WinForm窗体美化、WinForm窗体安全、WinForm窗体性能。



清华大学出版社

软件工程师培养丛书

TP316.7
325

WinForm 技术应用

武汉厚溥教育科技有限公司 编著

清华大学出版社

北 京

内 容 简 介

本书按照高等院校、高职高专计算机课程基本要求,以案例驱动的形式来组织内容,突出计算机课程的实践性特点。本书共包括9章:WinForm 基础、WinForm 窗体控件(一)、WinForm 窗体控件(二)、多文档窗体及控件布局、ADO.NET——数据库操作、ADO.NET——DataSet 和适配器、DataGridView——显示和操作数据、三层架构的应用以及面向对象实现三层架构。

本书附赠 PPT 教学课件和案例源文件,这些教学资源可通过<http://www.tupwk.com.cn/downpage>下载。

本书内容安排合理,层次清楚,通俗易懂,实例丰富,突出理论与实践的结合,可作为各类高等院校、高职高专及培训机构的教材,也可供广大 Windows 程序设计人员参考。

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

WinForm 技术应用 / 武汉厚溥教育科技有限公司 编著. —北京:清华大学出版社,2014
(软件工程师培养丛书)

ISBN 978-7-302-34749-1

I. ①W… II. ①武… III. ①Windows 操作系统 IV. ①TP316.7

中国版本图书馆 CIP 数据核字(2013)第 295222 号

责任编辑:刘金喜

封面设计:崔东方

版式设计:思创景点

责任校对:邱晓玉

责任印制:王静怡

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>, 010-62794504

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:21.5 字 数:369 千字

版 次:2014 年 1 月第 1 版 印 次:2014 年 1 月第 1 次印刷

印 数:1~3000

定 价:38.00 元

编 委 会

主 任:

翁高飞

副主任:

王 鹏	余晓刚	刘 伟	曹 静
方风波	邹治伟	李建利	管胜波

委 员:

罗 炜	耿 杰	谢日星	吴金秀
夏超群	陈 琴	夏 晶	彭 莉
徐 霞	明素华	王 敏	严 滔
胡智方			

前 言

Visual C#是微软公司.NET Framework 框架中的一个重要组成部分,也是微软公司极力推荐的新一代程序开发语言。WinForm 是.NET 开发平台中对 Windows Form 的一种称谓,.NET 为开发 WinForm 的应用程序提供了丰富的 Class Library(类库)。这些 WinForm 类库支持快速应用程序开发,这些类库被封装在一个命名空间之中,这个命名空间就是 System.Windows.Forms。在此命名空间中定义了许多类,在开发基于.NET 的 GUI 应用程序的时候,就是通过继承和扩展这些类使得我们的程序有着多样的用户界面。

本书隶属于“软件工程师培养丛书”中的一本专业基础教材,该丛书是由武汉厚溥教育科技有限公司开发,以培养符合企业需求的软件工程师应用开发、实施为目标的 IT 职业教育丛书。在开发该丛书之前,我们对 IT 行业的岗位序列做了充分的调研,包括研究从业人员技术方向、项目经验和职业素质等方面的需求,通过对面向的学生特点、行业需求的现状以及实施等方面的详细分析,结合“厚溥”对软件人才培养模式的认知,按照软件专业总体定位要求,进行软件专业产品课程体系设计。该丛书集应用软件知识和多领域的实践项目于一体,着重培养学生的熟练度、规范性、集成和项目能力,从而达到预定的培养目标。

本书是“软件工程师培养丛书”中《C#编程技术基础》一书内容的延续,在学习本书之前,请先学习《C#编程技术基础》。

本书共包括 9 章: WinForm 基础、WinForm 窗体控件(一)、WinForm 窗体控件(二)、多文档窗体及控件布局、ADO.NET——数据库操作、ADO.NET——DataSet 和适配器、DataGridView——显示和操作数据、三层架构的应用以及面向对象实现三层架构。

我们对本书的编写体系做了精心的设计,按照“理论学习—知识总结—上机操作—课后习题”这一思路进行编排。“理论学习”部分描述通过本案例要达到的学习目的与涉及的相关知识点,使学习目标更加明确;“知识总结”部分概括案例所涉及的知识,



使知识点完整系统地呈现；“上机操作”部分对案例进行了详尽分析，通过完整的步骤帮助读者快速掌握该案例的操作方法；“课后习题”部分帮助读者理解章节的知识点。本书在内容编写方面，力求细致全面；在文字叙述方面，注意言简意赅、重点突出；在案例选取方面，强调案例的针对性和实用性。

本书凝聚了编者多年来的教学经验和成果，可作为各类高等院校、高职高专及培训机构的教材，也可供广大 Windows 程序设计人员参考。

本书 PPT 教学课件和案例源文件可通过 <http://www.tupwk.com.cn/downpage> 下载。

本书由武汉厚溥教育科技有限公司编著，由翁高飞、余晓刚等多名企业实战项目经理编写。本书编者长期从事项目开发和教学实施，并且对当前高校的教学情况非常熟悉，在编写过程中充分考虑到不同学生的特点和需求，加强了项目实践方面的教学。本书在编写过程中，得到了武汉厚溥教育科技有限公司各级领导的大力支持，在此对他们表示衷心的感谢。

参与本书编写的人员还有武汉商学院曹静老师、荆州职业技术学院方风波老师、武汉工程职业技术学院邹治伟老师、湖北三峡职业技术学院李建利老师、武汉软件工程职业学院罗炜老师、湖北国土资源职业学院管胜波和王敏老师、黄冈职业技术学院夏晶老师、湖北三峡职业技术学院严滔老师等。

限于编写时间和编者的水平，书中难免存在不足之处，希望广大读者批评指正。

服务邮箱：wkservice@163.com。

编者

2013 年 10 月

目 录

第 1 章 WinForm 基础	1
1.1 Windows 窗体	2
1.1.1 Windows 窗体及其特点	2
1.1.2 创建 Windows 应用程序	3
1.1.3 Windows 应用程序的文件夹结构	6
1.1.4 通过应用程序编辑窗体	7
1.1.5 认识窗体重要属性	8
1.2 Windows 窗体基本控件	8
1.2.1 标签(Label)	8
1.2.2 文本框(TextBox)	9
1.2.3 按钮(Button)	10
1.2.4 列表框(ListBox)	12
1.3 使用消息框窗体	14
1.4 多个窗体的使用	16
1.5 综合实例	17
【小结】	22
【自测题】	22
【上机部分】	23
【课后作业】	28
第 2 章 WinForm 窗体控件(一)	31
2.1 菜单	32
2.1.1 菜单概述	32
2.1.2 创建菜单栏、菜单项	33
2.1.3 响应菜单事件	35
2.1.4 上下文菜单	36
2.2 托盘图标	39
2.3 工具栏	41
2.3.1 创建工具栏	41
2.3.2 添加工具项	42
2.4 状态栏	44
2.4.1 创建状态栏	44
2.4.2 添加状态栏项	45
【小结】	46
【自测题】	47
【上机部分】	47
【课后作业】	54
第 3 章 WinForm 窗体控件(二)	55
3.1 RadioButton、CheckBox 和 GroupBox	56
3.1.1 单选按钮和分组框	56
3.1.2 复选框 CheckBox 控件	59
3.1.3 组合框 ComboBox 控件	60



3.2 PictureBox、ImageList 和 Timer 控件	61
3.2.1 图片框控件	61
3.2.2 定时器 Timer 控件	62
3.2.3 图像列表	62
3.3 进度条 ProgressBar 控件	65
3.4 选项卡	66
3.5 控件布局	67
3.5.1 对控件进行分层	67
3.5.2 在窗体中定位控件	68
3.5.3 改变控件的大小	68
3.5.4 相对于窗体的边框固定控件	68
3.5.5 设置控件的 Tab 键顺序	69
【小结】	69
【自测题】	70
【上机部分】	70
【课后作业】	77
第 4 章 多文档窗体及控件布局	79
4.1 MDI 窗体	80
4.1.1 MDI 窗体概述	80
4.1.2 编写 MDI 窗体	81
4.1.3 MDI 窗体布局	83
4.1.4 MDI 窗体列表	87
4.2 TreeView 控件	92
4.3 Splitter 控件和 splitContainer 控件	95
4.4 Splitter、TreeView 控件综合示例	96
【小结】	106

【自测题】	106
【上机部分】	107
【课后作业】	115
第 5 章 ADO.NET——数据库操作	117
5.1 ADO.NET	118
5.1.1 ADO.NET 概述	118
5.1.2 简单地访问关系数据	118
5.1.3 可扩展性，支持更多的数据源	119
5.1.4 支持多层应用程序	119
5.1.5 ADO.NET 以 XML 为基础构建，扩展性强	120
5.2 .NET Framework 命名空间	120
5.2.1 .NET Framework 中的数据和 XML 命名空间	120
5.2.2 ADO.NET 的结构	121
5.3 .NET Framework 数据提供程序	121
5.3.1 SQL Server .NET Framework 数据提供程序	122
5.3.2 Oracle .NET Framework 数据提供程序	122
5.3.3 OLEDB .NET Framework 数据提供程序	123
5.3.4 ODBC .NET Framework 数据提供程序	123
5.3.5 SQL Server .NET Framework 数据提供程序和 OLE DB .NET Framework 数据提供程序的比较	123



5.4 .NET Framework 数据提供	
程序的核心对象	124
5.4.1 Connection 对象	125
5.4.2 Command 对象	125
5.4.3 DataReader 对象	126
5.4.4 DataAdapter 对象	126
5.5 ADO.NET 对数据库的	
访问	127
5.5.1 连接数据库	
(Connection)	127
5.5.2 执行 SQL 语句	
(Command)	132
5.6 ADO.NET 中的事务处理	151
5.6.1 事务说明	151
5.6.2 事务构建	152
5.6.3 Transaction 对象	152
【小结】	154
【自测题】	154
【上机部分】	155
【课后作业】	160

第 6 章 ADO.NET——DataSet 和	
适配器	161
6.1 DataSet	162
6.1.1 DataSet 概述	162
6.1.2 使用 C# 代码创建	
数据集	165
6.1.3 在 DataSet 中检索数据	170
6.1.4 DataSet 数据的 XML	
持久化处理	175

6.2 DataAdapter	178
6.3 综合演练——修改 MDI	
日记本	179
【小结】	187
【自测题】	187
【上机部分】	188
【课后作业】	195
第 7 章 DataGridView——显示和	
操作数据	197
7.1 DataGridView	198
7.1.1 控件概述	198
7.1.2 数据源(DataSource)	199
7.1.3 在 DataGridView 中添加、	
修改和删除信息	205
7.2 DataGridView 界面	
自定义	210
【小结】	214
【自测题】	214
【上机部分】	215
【课后作业】	223

第 8 章 三层架构的应用	225
8.1 分层设计	226
8.2 软件开发的分层	227
8.3 三层架构之间的关系——	
数据传递方向	228
8.4 搭建三层架构项目框架	229
8.4.1 表示层的搭建	230
8.4.2 业务逻辑层的搭建	230



8.4.3 数据访问层的搭建	230
8.5 综合案例	231
8.5.1 编写数据访问层代码	231
8.5.2 编写业务逻辑层代码	234
8.5.3 修改界面层代码	245
【小结】	247
【自测题】	248
【上机部分】	248
【课后作业】	254

第9章 面向对象实现三层架构

9.1 实体类	256
9.1.1 使用实体类的意义	256
9.1.2 实体类的概念	257

9.1.3 实体类的作用	261
9.2 使用实体类实现三层架构	262
9.2.1 新建实体类	262
9.2.2 添加每个层与实体层之间的引用关系	267
9.2.3 改造数据访问层	267
9.2.4 改造业务逻辑层	322
【小结】	324
【自测题】	325
【上机部分】	325
【课后作业】	331

第1章

WinForm基础



课程目标

- ▶ 了解窗体基础知识和常用属性
- ▶ 熟练使用标签、文本框、按钮控件
- ▶ 掌握列表框控件的使用
- ▶ 使用消息框弹出消息
- ▶ 熟练掌握窗体间链接



简介

在开始本书学习之前,请读者先学习“软件工程师培养丛书”中的《C#编程技术基础》,掌握 C#语言的基本语法,了解面向对象的基础语法——类和对象,以及面向对象的三大特点,即封装、继承和多态。在 C#编程技术的学习过程中,我们一直创建的是控制台应用程序,也就是字符控制台界面下的应用程序,从本书开始,我们将要学习如何创建图形用户界面(GUI)程序,也就是常见的 Windows 窗体应用程序。本章将使我们初步了解 Windows 应用程序的基础知识和一些基本控件的使用。

1.1 Windows 窗体

1.1.1 Windows 窗体及其特点

Windows 窗体,就是我们经常看到的用户界面“窗体”。在使用操作系统时,我们经常通过窗体来进行各种操作和设置,如设置桌面的分辨率大小、颜色、背景图片等,如图 1-1 所示。

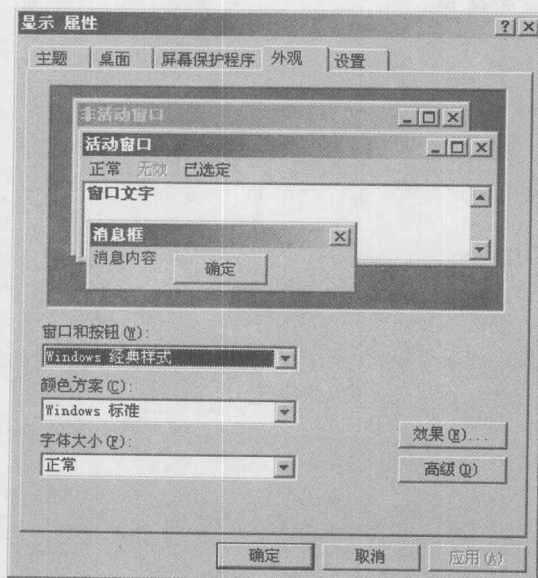


图 1-1



我们在操作计算机时，会经常使用类似这样的“窗体”。正因为有了这些“窗体”，我们操作计算机系统时才会非常简单、方便和灵活。也正因有了 Windows(窗体的复数，直译就是很多窗体的意思)操作系统，使得纯字符界面的单用户单任务 DOS 操作系统退出了历史舞台。

我们平时在操作计算机时，随时都会接触到不同的 Windows 窗体，这些窗体都具有类似的功能，如可以最大化、最小化和关闭等。而且，我们还会发现，窗体上的操作元素也经常重复出现，如信息输入框、按钮、下拉选项表和表格等。

以上都是 GUI(图形用户界面)的元素。对程序设计人员而言，使用 .NET Framework 提供的 Windows 窗体以及窗体控件，会让开发 Windows 窗体应用程序非常简单。Windows 窗体也简称为 WinForm，开发人员可以使用 C# 的“WinForm 应用程序项目”来创建应用程序的用户界面，编写少量代码就可以提供丰富的功能。

WinForm 应用程序一般都有一个或者多个窗体提供用户与应用程序交互。窗体可包含文本框、标签、按钮等控件。一般的 WinForm 应用程序有许多窗体，有的是获得用户输入的数据，有的是向用户显示数据，有的窗体具有变形、透明等特殊效果，让用户不知道它的存在。比如 QQ 中鼠标指向一个好友头像时，弹出的悬浮信息就是一个窗体；又比如迅雷或者其他下载工具的悬浮窗，其实也是一个窗体，尽管这两者在外观上与常见的“窗体”不同，但实际上，它们都是同一种东西。

System.Windows.Forms 命名空间里定义了创建 WinForm 应用程序时所需的类。Windows 窗体的一些重要特性如下。

- 简单强大的功能：可以用于设计窗体和可视控件，创建丰富的基于 Windows 的图形界面应用程序。
- 丰富的控件：Windows 窗体提供了一套丰富的控件，并且开发人员可以定义自己特色的新控件。
- 快捷的数据显示和操作：应用程序开发中最常见的情形之一是在窗体上显示数据。Windows 窗体对数据库处理提供全面支持，快速访问数据库中的数据，并在窗体上显示和操作数据。(本书重点)

1.1.2 创建 Windows 应用程序

我们现在创建第一个 Windows 应用程序，用 C# 创建应用程序的步骤如下：



(1) 单击“开始”|“程序”|“Microsoft Visual Studio 2008”|“Microsoft Visual Studio 2008”。

(2) 单击“文件”|“新建”|“项目”，此时显示“新建项目”对话框，如图 1-2 所示。

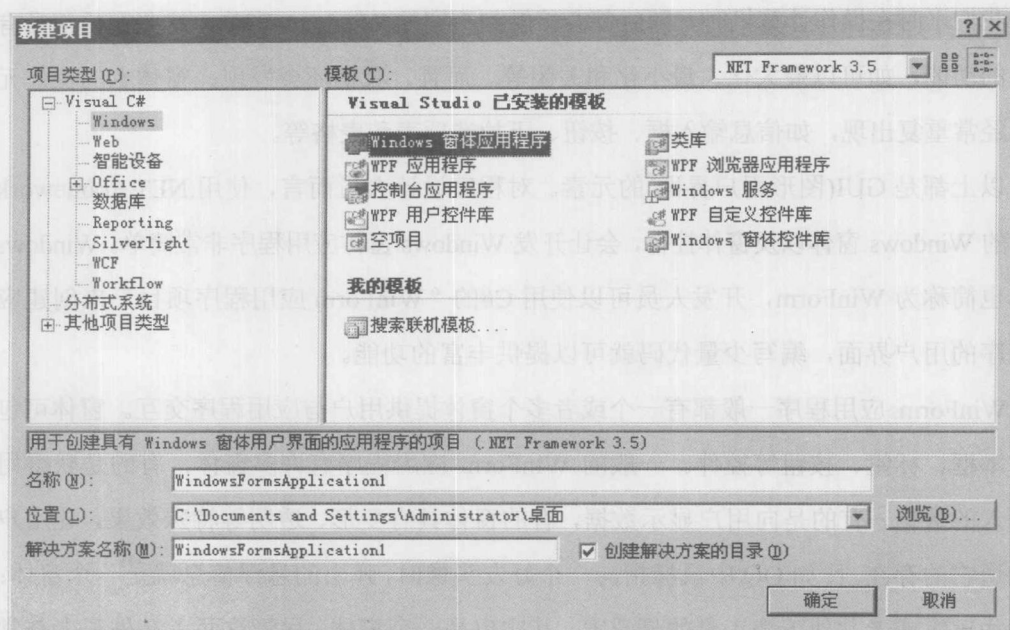


图 1-2

在“项目类型”列表中单击“Visual C#”前的“+”，选择“Windows”，然后在模板中选择“Windows 窗体应用程序”。在下面的“名称”栏中输入应用程序的名称，在“位置”栏中选择应用程序所放的位置并单击“确定”按钮。完成后，会显示如图 1-3 所示的编辑界面。

此时我们看到，在 VS 2008(Visual Studio 2008)环境里显示的组成结构和控制台应用程序的不同。首先正中间是一个空白窗体，等着我们去编辑，如图 1-4 所示。在空白窗体的左侧会看到工具箱栏，在工具箱栏里，列出了 Windows 窗体常用控件，如图 1-5 所示。我们可以将工具箱里的控件添加到空白窗体里(后面章节将会介绍)。

在空白窗体右侧有两个窗体栏，右上边是解决方案资源管理器，作用是管理本项目的程序文件；右下边是属性栏窗体(作用后面会学习到)。讲到这里，我们现在执行一下这个程序，虽然没有填写一行代码，但是 VS 环境已经自动生成了很多代码，可以显示这个空白窗体。按下 F5 键后，程序执行后显示如图 1-4 所示，实际运行效果和设计器



效果一模一样。正是这种所见即所得的编写方式，让编写窗体程序不再是 C/C++ 时代那么令人痛苦的工作。

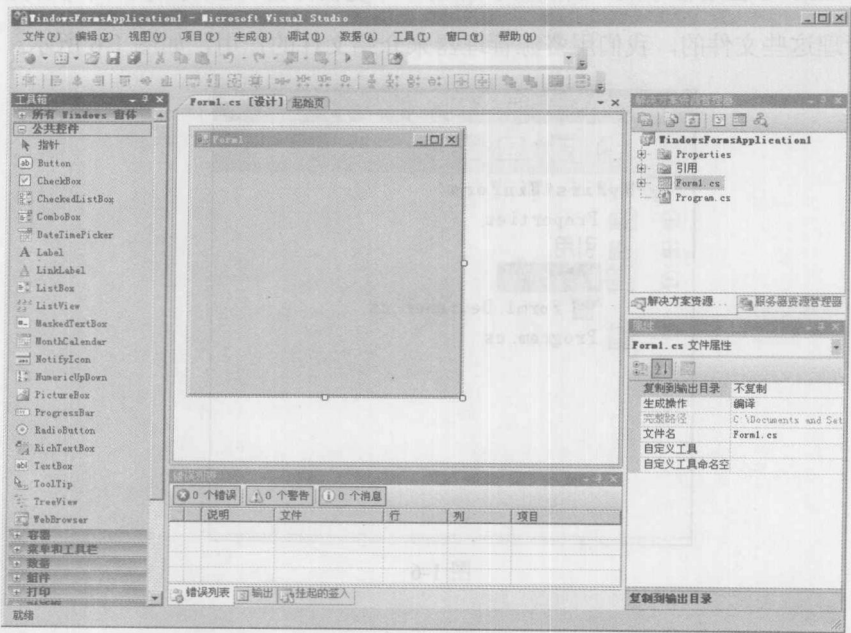


图 1-3

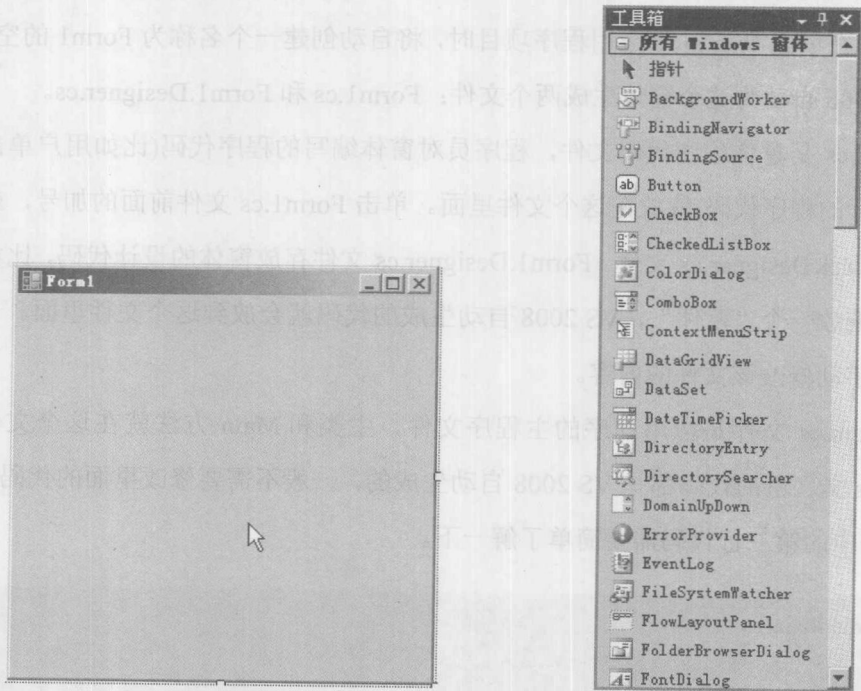


图 1-4

图 1-5



1.1.3 Windows 应用程序的文件夹结构

创建 Windows 应用程序后,当然要了解编译环境软件自动生成的文件和文件夹。资源管理器是管理这些文件的,我们用资源管理器来介绍文件的作用,如图 1-6 所示。



图 1-6

在这里,我们看到了和控制台应用程序相似的文件结构,它实际上是一个树状结构。根节点是解决方案名称,接下来的子节点是项目名称,再接下来第三层次就是项目中的程序文件。在创建 Windows 应用程序项目时,将自动创建一个名称为 Form1 的空窗体,并且 VS 2008 自动为这个窗体生成两个文件: Form1.cs 和 Form1.Designer.cs。

Form1.cs 是窗体程序代码文件,程序员对窗体编写的程序代码(比如用户单击一个按钮后执行的程序代码)就放在这个文件里面。单击 Form1.cs 文件前面的加号,就会展开看到 Form1.Designer.cs 文件。Form1.Designer.cs 文件存放窗体的设计代码,比如我们往窗体上拖放一个“控件”,VS 2008 自动生成的代码就会放到这个文件里面。一般情况下不要手动修改该文件的内容。

Program.cs 文件是应用程序的主程序文件,主类和 Main 方法就在这个文件中。Program.cs 文件里的代码都是 VS 2008 自动生成的,一般不需要修改里面的代码。其中 Main 方法中的第三行代码需要简单了解一下。

```
static void Main()
```

```
{
```

```
//允许使用 XP 风格的窗体样式(去掉看看是什么效果)
```