



C#.NET 桌面应用项目 开发教程

李伟 汪剑 姜淋译 / 主 编
崔丽 牟奇春 姜政波 王一棕 刘秀莲 / 副主编

清华大学出版社





C#.NET 桌面应用项目 开发教程



李伟 汪剑 姜淋译 / 主 编

崔丽 牟奇春 姜政波 王一棕 刘秀莲 / 副主编

清华大学出版社
北 京

内 容 简 介

本书以企业管理信息系统的开发为主线,以 Visual Studio 2010 作为开发工具,采用循序渐进的方式系统地介绍使用 C# 语言进行项目开发所涉及的关键知识。

全书共分为 6 章。其中,第 1、2 章通过企业管理信息系统的实现过程介绍 C# 语言及其开发环境、C# 项目背景和系统分析及程序设计的基础知识;第 3 章以企业管理信息系统的建模与设计的实现过程为例,介绍 C# 面向对象编程的相关知识;第 4 章以信息系统关键模块的设计与实现深入介绍利用 C# 进行 Windows 编程的相关知识,包括控件的使用、ADO.NET 数据库访问技术以及 Windows 应用程序的部署;第 5 章介绍 C# 语法与索引的相关知识;第 6 章介绍开发技术及数据库操作。

本书内容新颖、结构安排合理、案例丰富实用,既可以作为高等学校计算机及其相关专业的教材,也可以作为相关培训机构和软件开发人员的参考书。

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#.NET 桌面应用项目开发教程/李伟,汪剑,姜淋译主编. —北京:清华大学出版社,2016
ISBN 978-7-302-41252-6

I. ①C… II. ①李… ②汪… ③姜… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 186392 号

责任编辑:焦 虹

封面设计:傅瑞学

责任校对:徐俊伟

责任印制:刘海龙

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

印装者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:12

字 数:278 千字

版 次:2016 年 1 月第 1 版

印 次:2016 年 1 月第 1 次印刷

印 数:1~2000

定 价:25.00 元

产品编号:062690-01

在信息技术历史上,大约每 10 年,新的程序设计方法就会像浪潮一样袭来。20 世纪 80 年代早期,新技术是可以运行在桌面上的 UNIX 系统以及 AT&T 开发的强大的 C 语言。20 世纪 90 年代初,Windows 和 C++ 占据了程序设计领域的半壁江山。这些发展推动着程序设计方法的进步,.NET 和 C# 代表的正是下一次浪潮。本书将帮助读者成为浪尖上的弄潮儿。

C# 作为微软公司专为 .NET 平台量身定做的编程语言,建立在 C(高性能)、C++(面向对象的结构)、Java(安全性)和 VB(快速开发)等语言的众多经验之上,在桌面应用、Web 应用程序、RIA 应用程序和智能手机应用程序等多个领域都显示出强大的功能,目前已成为主流的开发工具。为了帮助读者掌握 .NET 环境下的程序开发技术,提高项目开发能力,编者结合 .NET 技术的开发实践以及多年 .NET 程序设计教学的经验,编写了本书。

全书以企业管理信息系统的开发与实现为示例来讲解 C# 程序设计的知识,采用项目驱动的教学方法,把一个完整的项目分解成不同的单元,并分散到各个章节中。通过“提出问题(做什么)—解决问题(跟我做)—问题探究(为什么这样做)—拓展与提高”的思路来完成每一个单元的设计与实现,最后形成一个完整的系统进行发布、部署,突出项目开发能力的训练,注重培养学生的系统观念和实用技能。本书具有以下特点:

(1) 内容实用,针对性强。以企业对 .NET 开发人员所要求的知识和技能来安排和组织内容,由浅入深地介绍了 .NET 开发人员必备的 C# 程序设计的基本知识和技能。

(2) 项目驱动,体现“做中学”的思想。每章内容均由案例引出,然后给出案例的实现过程,介绍相关知识点,最后总结提高。读者通过模仿、探究和提升,能够提高学习兴趣,真正掌握 C# 程序设计的方法。

(3) 以实际项目为依托。将企业管理系统的开发分解成不同的模块,按照循序渐进的原则分散到不同的章节进行设计实现,最后再

集成发布,使读者能够将所学知识综合应用,提高项目开发能力。

(4) 配套教学资源丰富。提供真实、完整的教学课件以及所有实例的源代码,方便读者学习和教学。

本书在编写过程中,参考了大量相关书籍和网络资源,在此对相关作者表示感谢。由于知识和时间的限制,本书难免会有一些不如意的地方,恳请读者批评指正。

配套教学资源网站地址 <http://zyk.cdp.edu.cn:10100>。

编 者

目录

Contents

第 1 章	开发环境介绍与应用	1
1.1	.Net 集成开发环境 Microsoft Visual Studio 2010	1
1.2	开发环境 Microsoft Visual Studio 2010 介绍	4
1.3	创建 Windows 应用程序	8
1.4	运行程序	13
第 2 章	健身俱乐部管理系统介绍	15
2.1	系统背景与目的	15
2.2	系统分析	15
2.2.1	需求分析	15
2.2.2	可行性分析	16
2.2.3	项目计划书编制	17
第 3 章	健身俱乐部管理系统建模与设计	19
3.1	系统设计	19
3.1.1	系统功能结构	19
3.1.2	系统业务流程	20
3.2	程序代码设计	26
3.2.1	编码规范	26
3.3	数据库设计	27
3.3.1	数据库概念设计	27
3.3.2	数据库逻辑设计	29
3.3.3	数据库创建	32
第 4 章	健身俱乐部管理系统实现	34
4.1	系统预览	36



4.2	公共类的抽取与实现	38
4.2.1	数据库操作公共类	38
4.2.2	统一异常处理公共类	47
4.3	登录模块实现	51
4.3.1	登录模块概要设计	51
4.3.2	登录模块界面与布局	52
4.3.3	Button、Label、TextBox 控件	52
4.3.4	核心代码设计	55
4.3.5	单元测试	67
4.3.6	实战经验之模板设计模式	70
4.3.7	模块实践	71
4.4	主窗体实现	71
4.4.1	主窗体概要设计	71
4.4.2	主窗体界面与布局	72
4.4.3	MenuStrip、ToolStrip、Timer 控件	72
4.4.4	核心代码设计	77
4.4.5	实战经验之单例模式	81
4.4.6	模块实践	82
4.5	消费单据模块实现	83
4.5.1	消费单据概要设计	83
4.5.2	消费单据界面与布局	83
4.5.3	DockContent、ComboBox、DataGridView 控件	83
4.5.4	核心代码设计	88
4.5.5	实战经验之简单工厂模式	103
4.5.6	模块实践	106
4.6	会员管理模块实现	106
4.6.1	会员管理概要设计	106
4.6.2	会员管理界面与布局	107
4.6.3	ContextMenuStrip、PrintDocument、PrintPreviewDialog 控件	108
4.6.4	核心代码设计	111
4.6.5	实战经验之多线程	122
4.6.6	模块实践	124
4.7	统计模块实现	124
4.7.1	统计报表模块概要设计	124
4.7.2	统计报表界面实现与布局	125
4.7.3	DataSet、RDLC、ReportViewer 控件	126
4.7.4	核心代码设计	129
4.7.5	实战经验之 MVC 模式	134

4.7.6 模块实践	135
第5章 C#基础语法及索引	136
5.1 类的基础	136
5.1.1 声明类	136
5.1.2 访问修饰符	137
5.1.3 类成员	137
5.1.4 类的字段	137
5.1.5 类的属性	138
5.1.6 类的实例和访问类成员	140
5.1.7 this 关键字	141
5.2 方法	141
5.2.1 方法的结构	141
5.2.2 方法的调用	142
5.2.3 参数(形参和实参)	143
5.2.4 方法重载	143
5.3 类和继承	144
5.3.1 类继承	144
5.3.2 抽象类	145
第6章 常用相关开发技术	147
6.1 关于事件	147
6.2 数据库操作	149
6.3 皮肤控件	150
6.3.1 皮肤控件的添加	150
6.3.2 皮肤控件的使用	151
6.4 Microsoft Visual Studio 2010 安装包制作	153
6.5 复合控件与自定义控件	158
6.5.1 复合控件	158
6.5.2 自定义控件	162
附录A Visio 的安装与使用	167
附录B PowerDesigner 的安装与使用	171
附录C C#关键字	182

开发环境介绍与应用

1.1 .Net 集成开发环境 Microsoft Visual Studio 2010

本书以 Microsoft Visual Studio 2010 作为开发平台,在安装 Microsoft Visual Studio 2010 之前,首先确保 IE 浏览器版本为 6.0 或更高且关闭当前打开的所有应用程序。

(1) 将 Microsoft Visual Studio 2010 安装光盘插入光驱,双击安装光盘中的 Setup.exe 安装文件。启动安装文件后会进入到 Microsoft Visual Studio 2010 安装界面,如图 1.1 所示。



图 1.1 Microsoft Visual Studio 2010 安装界面

(2) 在 Microsoft Visual Studio 2010 安装界面出现两个安装选项: Install Microsoft Visual Studio 2010(安装 Microsoft Visual Studio 2010)和 Check for Service Releases(检查 Service Releases)。通常情况下,单击 Install Microsoft Visual Studio 2010 进行 Microsoft Visual Studio 2010 安装。在进行 Microsoft Visual Studio 2010 安装前会加载安装组件。加载这些组件的目的是为了保证后续的安装顺利进行,必须等待组件加载完成后才能继续下一步。在组件加载界面的左侧存在一个选择框提示用户是否将该软件的经验信息发送给微软,此选项对实际用户来说可选可不选,如图 1.2 所示。

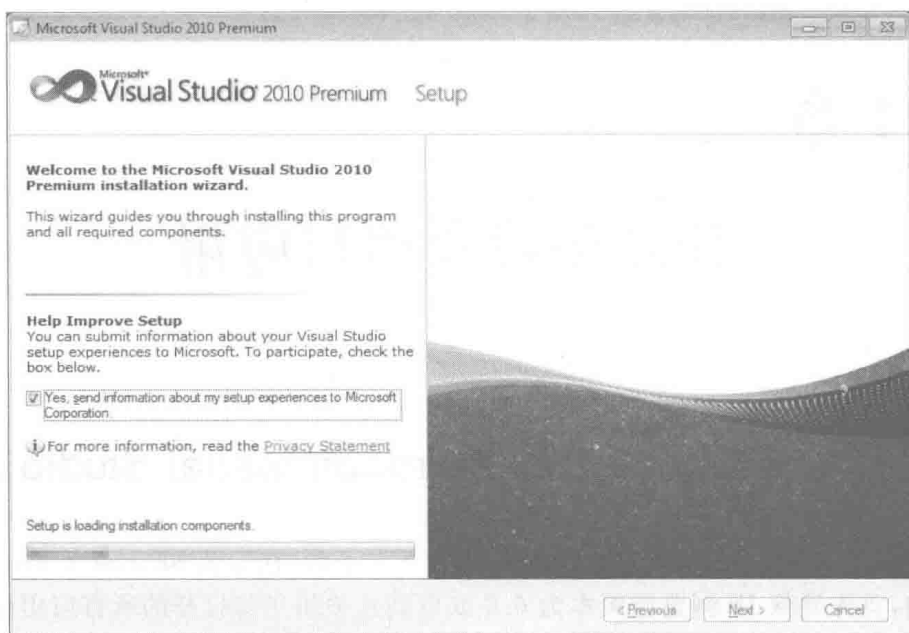


图 1.2 Microsoft Visual Studio 2010 组件加载界面

(3) 在安装组件加载完成后,单击 Next(下一步)按钮进入接受 license(软件许可)界面。在这个界面中选择 I have read and accept the license terms(已阅读并且接受条款)选项,如图 1.3 所示。

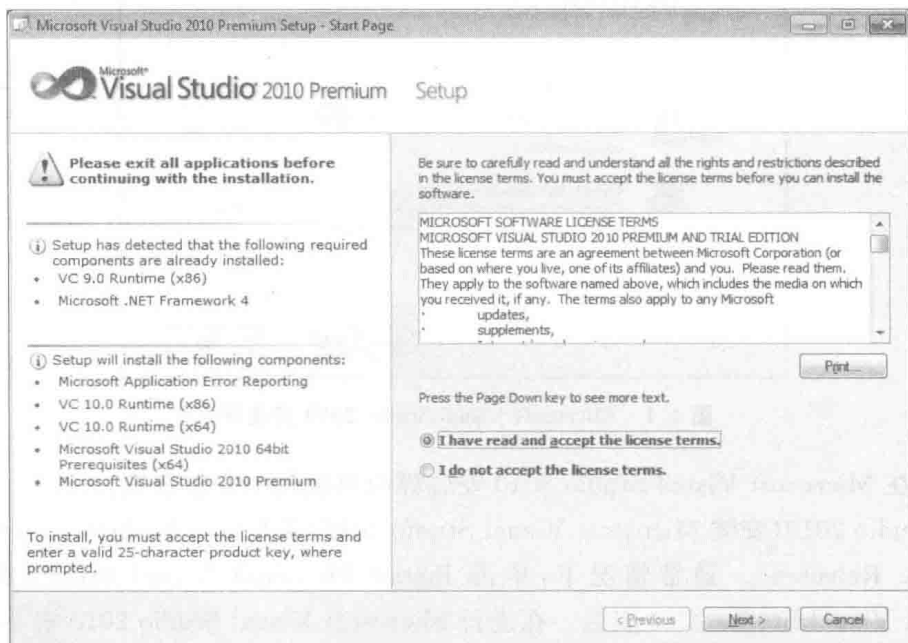


图 1.3 接受 license(软件许可)界面

(4) 单击 Next(下一步)按钮开始安装 Microsoft Visual Studio 2010。在这个界面里,可以指定 Microsoft Visual Studio 2010 的安装路径,如图 1.4 所示。除此之外该界面提供了相

应的安装模式,可以选择 Full(完全)或者 Customer(自定义)安装。选择 Full(完全)将安装 Microsoft Visual Studio 2010 所支持的全部编程语言及工具。选择 Custom(自定义)将安装用户需要的部分编程语言及工具。在此推荐选择 Full(完全)进行安装。

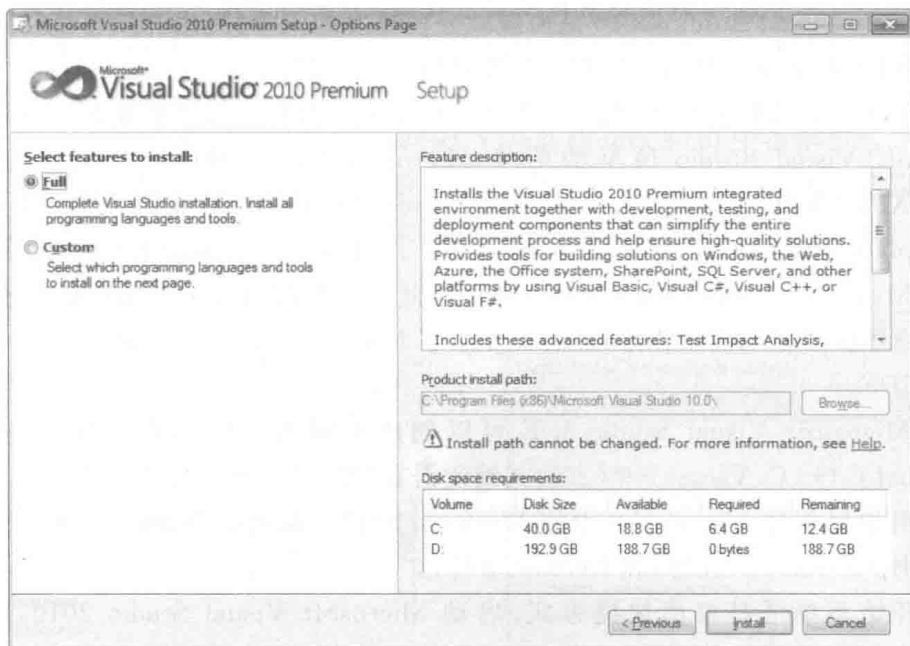


图 1.4 Microsoft Visual Studio 2010 安装模式及安装路径

(5) 单击 Install(安装)按钮开始 Microsoft Visual Studio 2010 的安装,安装界面中出现了安装列表及安装进度条,如图 1.5 所示。安装完成后会出现安装成功界面,至此 Microsoft Visual Studio 2010 在用户本地计算机中安装成功。

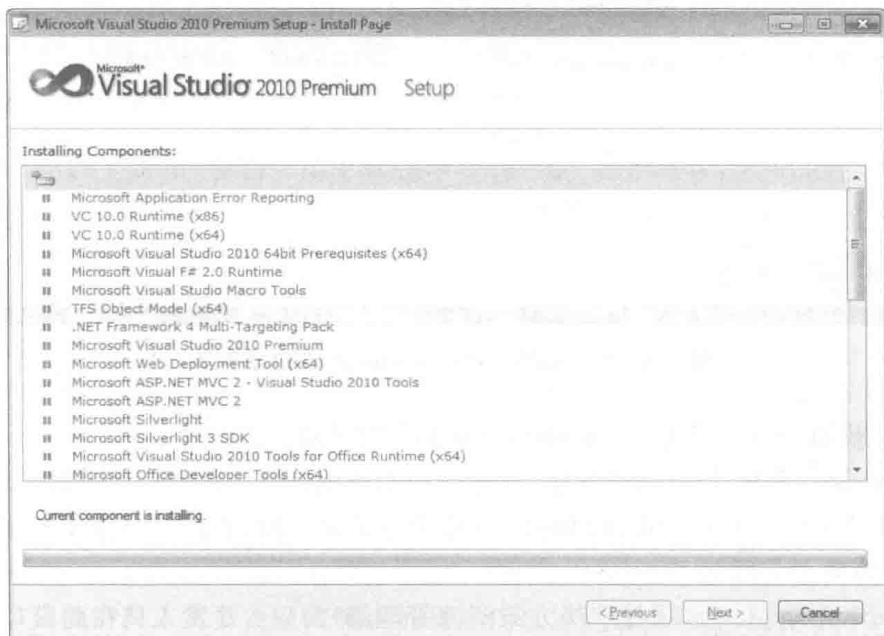


图 1.5 Microsoft Visual Studio 2010 的安装

1.2 开发环境 Microsoft Visual Studio 2010 介绍

Microsoft Visual Studio 2010 是由微软公司开发的 IDE(集成开发环境),该 IDE 是目前最流行的 Windows 平台应用程序开发环境。

在该版本之前微软也公布过诸多该 IDE 的前身。1998 年,微软发布了其第一个也是 Microsoft Visual Studio 最早的前身 Microsoft Visual Studio 6.0。2002 年随着 Windows XP 的发布,微软发布了 Microsoft Visual Studio 2002;在随后的一年里又发布了 Microsoft Visual Studio 2003;2005 年又发布了 Microsoft Visual Studio 2005;2008 年又发布了 Microsoft Visual Studio 2008,该版本的 IDE 使得开发人员能够更高效、更快速地创建用户体验丰富的应用程序;其后又发布了 Microsoft Visual Studio 2010,也就是现在使用的 IDE。

利用 Microsoft Visual Studio IDE 可以创建不同语言的应用程序,例如 Visual Basic、Visual C++、C、Visual F#(方法式编程语言)等;当然也可以创建 Visual C#。本书主要介绍如何创建 Visual C# 中的 Winform 项目。典型的 Winform 项目通常包括 Forms(窗体)、Controls(控件)和 Events(事件)。

通过开始菜单或是桌面快捷方式,启动 Microsoft Visual Studio 2010,如图 1.6 所示。

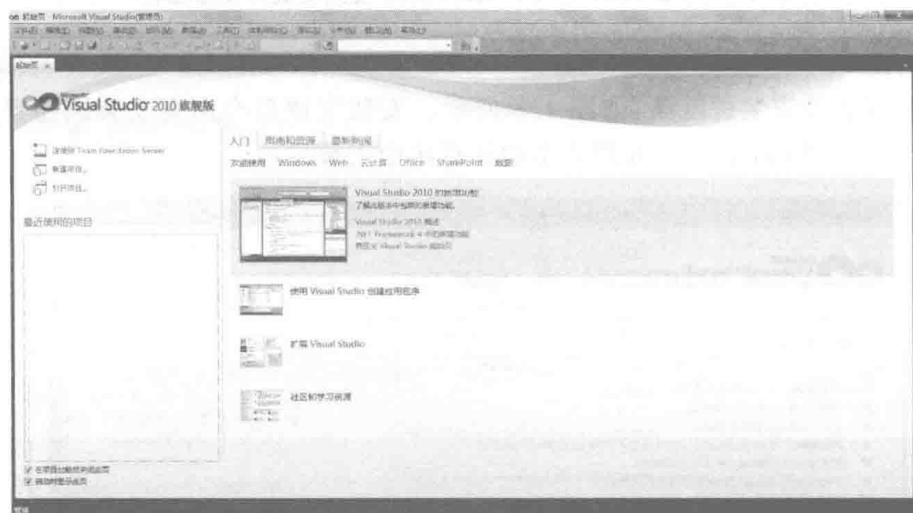


图 1.6 Microsoft Visual Studio 2010 启动界面

首先介绍 Microsoft Visual Studio 2010 的开发环境。

- Menubar(菜单栏): 菜单栏位于 Microsoft Visual Studio 2010 主窗口的顶部,包含了 Microsoft Visual Studio 2010 的所有功能,可以直接通过菜单栏实现各种操作,如图 1.7 所示。
- Solution Explorer(项目解决方案资源管理器)窗口: 开发人员在此窗口中选择相应文件,双击文件后代码会出现在主窗口中。该窗口可以显示当前打开的工程所



图 1.7 菜单栏界面

包含的窗体、类和组件。例如：在通常应用开发中需要不同开发人员进行不同组件的开发,项目解决方案窗口便可以解决此问题。一个解决方案下有多个项目或多个类库时,如图 1.8 所示。

- Properties(属性)窗口：在 Microsoft Visual Studio 2010 中提供了给开发人员进行应用开发的控件。每个控件都有属性,通过不同控件属性的配置实现不同的功能。属性窗口用于显示目前选择对象的属性和设置该对象的事件响应方法,如图 1.9 所示。



图 1.8 项目解决方案窗口



图 1.9 属性窗口

- ToolBox(工具箱)窗口：在工具箱中包含 Microsoft Visual Studio 2010 对 .Net 应用程序支持的控件。开发人员利用工具箱中的控件进行应用程序开发。在此窗口中可以找到 Winform 编程中常用的控件,使用时直接拖曳里面的控件到设计器窗体上即可,如图 1.10 所示。
- ToolBar(工具栏)界面：提供常用的快捷功能按键。例如：在 Debug 时要用到的执行、单步执行等,如图 1.11 所示。
- Error List(错误列表窗口)：在应用程序开发中的错误都会体现在此窗口中,包括应用程序在编译时出现的错误信息、警告消息、消息选项卡。若开发过程中出现错误信息而不处理,程序将编译失败,从而使程序无法成功运行。通常情况下双击错误信息列表中的任意一条信息,可以跳转到相应的错误代码位置,方便修正。其他情况下需要通过该错误的详细信息才能解决问题,如图 1.12 所示。
- Designer(设计器窗口)：该窗口显示窗体的设计界面。在窗体的设计期间,可以将工

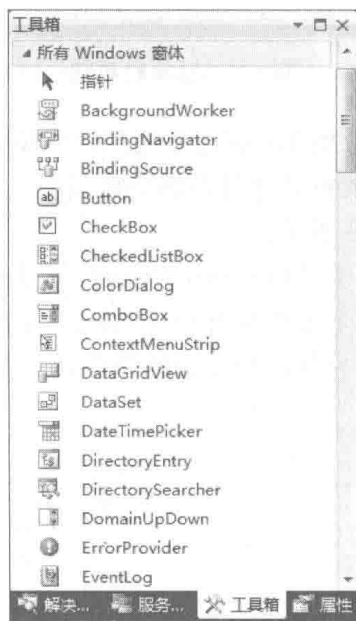


图 1.10 工具箱窗口



图 1.11 工具栏界面

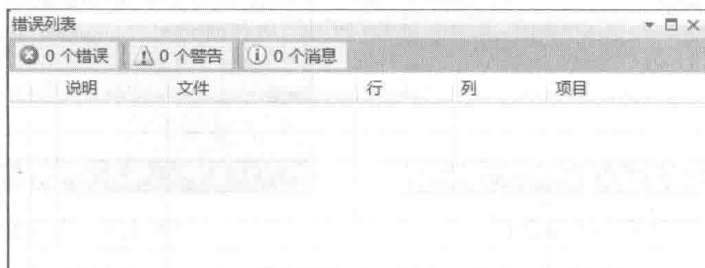


图 1.12 错误列表窗口

具箱中的控件直接拖放到当前窗体的设计器窗口中。设计器窗口如图 1.13 所示。

- Code(代码窗口)：该窗口显示当前窗体的后台代码。后台代码就是界面上相关的后台逻辑。通过后台代码可以实现对窗体的任何操作,包括窗体中控件的显示与隐藏、窗体出现的位置、窗体的大小等。代码窗口如图 1.14 所示。

在应用程序开发时,有时避免不了将有些窗体关闭。例如:有时把“错误列表”关闭了,又需要打开时该怎么处理呢?可以通过工具栏中的一些功能将其还原。比如“错误列表”窗口或是“属性”窗口被错误关闭,可以在工具栏中单击错误列表窗口或属性窗口,此时被关闭的窗口将会回到原来的位置。如果想要恢复的窗口在工具栏中没有,就可以在 View(视图)菜单中打开。如果 View(视图)菜单中也没有,还可以查看 View(视图)下的 Other Window(其他窗口)的菜单项,如图 1.15 所示。

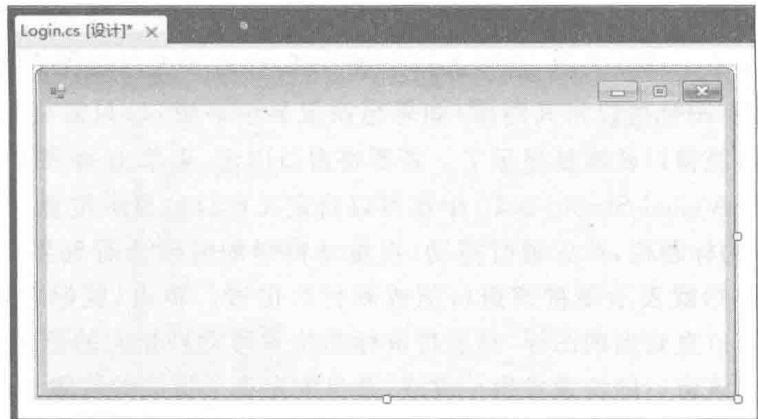


图 1.13 设计器窗口

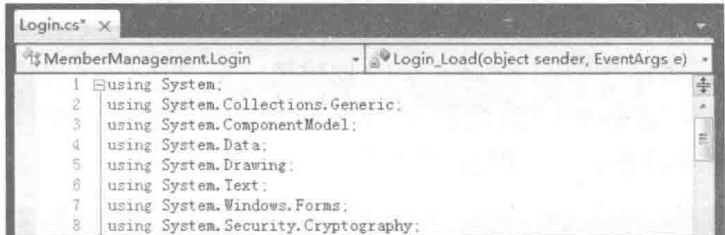


图 1.14 代码窗口

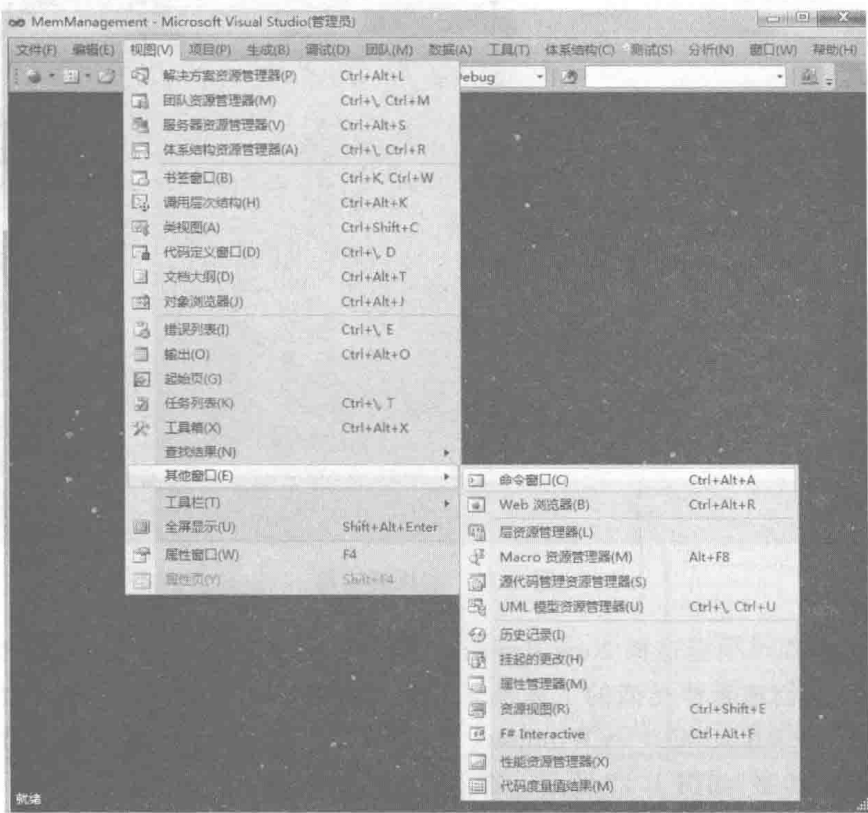


图 1.15 恢复关闭的窗口

Microsoft Visual Studio 2010 的用户体验非常丰富。有时一个窗口显示在工具中有些占位置,但是又不想将其关闭,这时选择隐藏即可,在需要的时候再恢复显示。单击该窗口标题栏上的  图标可以将其隐藏,如果想恢复显示该窗口,只需将鼠标移动到该窗口的标签上,这时该窗口就恢复显示了。若要将窗口固定,则单击  图标。

在 Microsoft Visual Studio 2010 中也可以自定义窗口的显示位置,操作很简单。只需要单击该窗口的标题栏,然后进行拖动,在拖动窗口的时候会看到界面上出现各种不同的图标,这些图标就表示要将该窗口拖放到什么位置。单击(鼠标左键)要拖动的窗口,一直拖动该窗口直到出现图标,然后将鼠标的位置移动到相应的图标上,释放鼠标左键,这时可以看到该窗口的位置发生了改变,并且固定在了拖放的位置。

1.3 创建 Windows 应用程序

本节使用 Microsoft Visual Studio 2010 创建第一个 Windows 应用程序。

(1) 创建一个 Winform 工程,启动 Microsoft Visual Studio 2010 后,单击菜单栏上的 File(文件)菜单项,如图 1.16 所示。

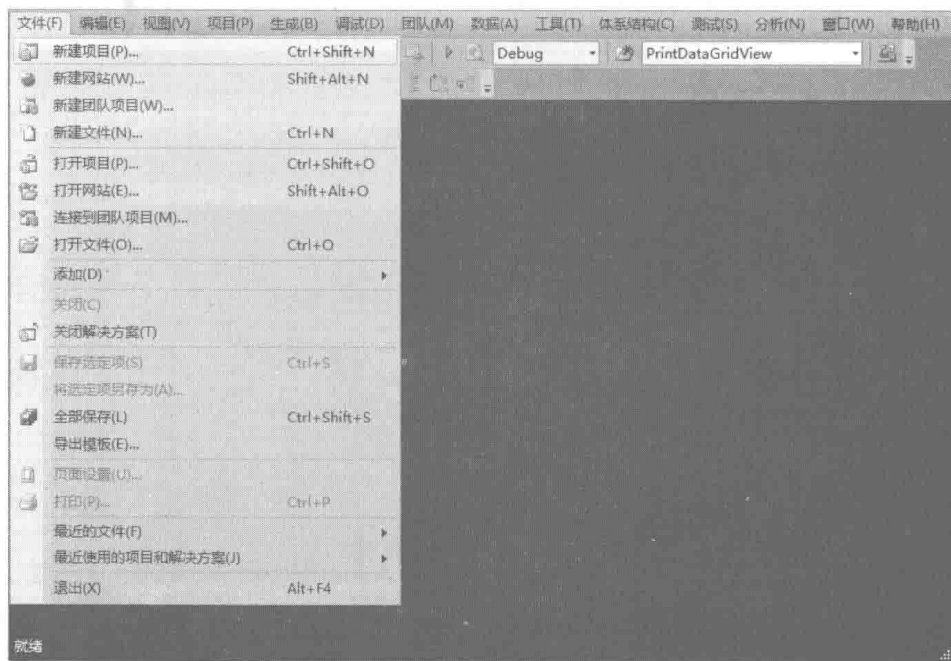


图 1.16 创建新工程

(2) 在下拉菜单项里选择 New Project(新建项目)命令后弹出一个对话框。在这个对话框中选择要创建哪种类型的工程及工程存放路径。这里选择 Visual C# 语言下的 Windows 工程 → Visual C# → Windows Forms Application(窗体应用程序),并输入 Demo 作为工程名字,如图 1.17 所示。在 Microsoft Visual Studio 2010 中除了可以创建 Windows 应用程序外还可以创建其他程序,包括 WPF 应用程序以及控制台应用程序等。

(3) 单击“确定”按钮后将成功创建一个 Windows 应用程序,在 Microsoft Visual



图 1.17 创建工程界面

Studio 2010 的窗体设计器中包含一个默认名为 Form1 的 Windows 空白窗体,如图 1.18 所示。窗体左侧是 Solution Explorer(解决方案资源管理器)窗口,在这个窗口里显示着整个工程的源代码文件以及资源文件等。窗体右侧是 ToolBox(工具箱)窗口,在这里有微软标准的常用控件可供选择。Microsoft Visual Studio 2010 允许用户通过鼠标将 Solution Explorer(解决方案资源管理器)和 ToolBox(工具箱)窗口进行拖放操作。可以根据每个人的开发习惯将这些窗口拖放到 Microsoft Visual Studio 2010 中不同的位置。例如,将图 1.18 中的 Solution Explorer(解决方案资源管理器)窗口与 ToolBox(工具箱)窗口进行左右互换。

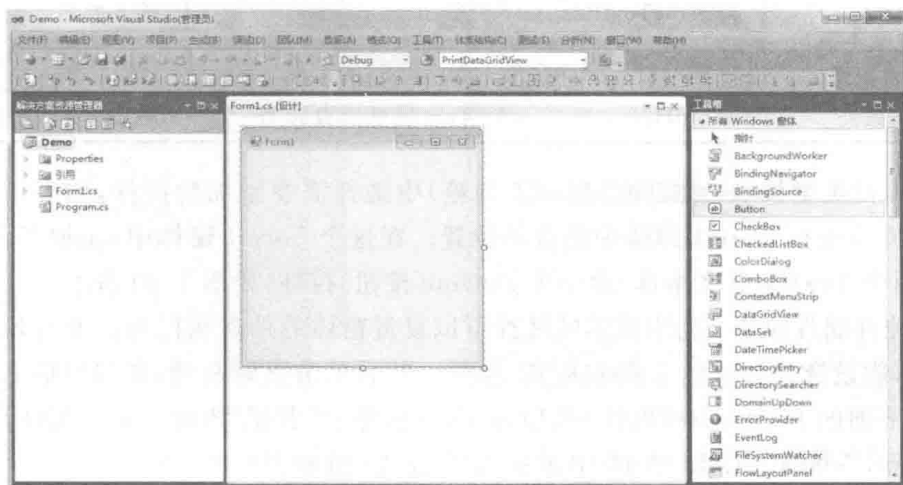


图 1.18 创建工程后的主界面