普通高等教育计算机规划教材

C#程序设计教程

刘 军 刘瑞新 主编
朱 立 张治斌 张 凡 等编著

提供电子教案

下载网址 http://www.cmpedu.com

机械工业出版社
CHINA MACHINE PRESS

普通高等教育计算机规划教材

C#程序设计教程

刘 军 刘瑞新 主编

朱 立 张治斌 张 凡 等编著

机械工业出版社

本书以 Visual Studio 2008/2005 为开发平台，以 C#为开发语言，面向无程序设计基础的读者，采用“任务驱动”方式，全面细致地介绍了 C#程序设计语言的基础知识、特点和具体应用。本书将面向对象的思想贯穿于整个教材，不仅在讲述内容上详细介绍了面向对象的相关概念及编程技巧，而且在所有例题、习题及上机实训中采用“任务驱动”的方式，强调使用面向对象的程序设计方法实现程序功能。强调程序功能由类及其属性、方法等实现。本书共分为 12 章，主要包括 C#语法基础，流程控制语句与控件，面向对象程序设计方法，数组、结构与集合，接口、委托和事件，泛型，异常处理、程序调试和文件操作，数据库操作，LINQ to SQL 数据库操作等内容。

本书适合作为高等学校、软件学院、职业院校的计算机及相关专业的教材，也适用于从事软件开发和应用的人员参考。

需要本书配套电子课件的教师可登录 www.cmpedu.com 免费注册、审核通过后下载，或联系编辑索取（QQ：1239258369，电话：010-88379739）。

图书在版编目（CIP）数据

C#程序设计教程 / 刘军，刘瑞新主编. —北京：机械工业出版社，2012.4
普通高等教育计算机规划教材
ISBN 978-7-111-38149-5

Ⅰ. ①C… Ⅱ. ①刘…②刘… 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（2012）第 078185 号

机械工业出版社（北京市百万庄大街 22 号　邮政编码 100037）
责任编辑：郝建伟　和庆娣
责任印制：乔　宇
三河市宏达印刷有限公司印刷
2012 年 8 月第 1 版 • 第 1 次印刷
184mm×260mm • 19 印张 • 470 千字
0001 - 3000 册
标准书号：ISBN 978-7-111-38149-5
定价：39.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务
社服务中心：（010）88361066
销售一部：（010）68326294
销售二部：（010）88379649
读者购书热线：（010）88379203

网络服务
门户网：http://www.cmpbook.com
教材网：http://www.cmpedu.com

出 版 说 明

信息技术是当今世界发展最快、渗透性最强、应用最广的关键技术，是推动经济增长和知识传播的重要引擎。在我国，随着国家信息化发展战略的贯彻实施，信息化建设已进入了全方位、多层次推进应用的新阶段。现在，掌握计算机技术已成为21世纪人才应具备的基础素质之一。

为了进一步推动计算机技术的发展，满足计算机学科教育的需求，机械工业出版社聘请了全国多所高等院校的一线教师，进行了充分的调研和讨论，针对计算机相关课程的特点，总结教学中的实践经验，组织出版了这套“普通高等教育计算机规划教材”。

本套教材具有以下特点：

1）反映计算机技术领域的新发展和新应用。

2）为了体现建设“立体化”精品教材的宗旨，本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、多媒体光盘、课程设计和毕业设计指导等内容。

3）针对多数学生的学习特点，采用通俗易懂的方法讲解知识，逻辑性强、层次分明、叙述准确而精炼、图文并茂，使学生可以快速掌握，学以致用。

4）符合高等院校各专业人才的培养目标及课程体系的设置，注重培养学生的应用能力，强调知识、能力与素质的综合训练。

5）注重教材的实用性、通用性，适合各类高等院校、高等职业学校及相关院校的教学，也可作为各类培训班和自学用书。

希望计算机教育界的专家和老师能提出宝贵的意见和建议。衷心感谢计算机教育工作者和广大读者的支持与帮助！

机械工业出版社

前 言

Visual C#是微软公司 Visual Studio 开发平台中提供的完全面向对象的编程语言。利用这种面向对象的、可视化的编程技术，结合事件驱动的模块设计，将使程序设计变得轻松快捷。因此，它在国内外各个领域中得到了广泛的应用。本书结合大量易于理解的实例，面向无编程基础的读者逐步学习 Visual C#程序设计的整个过程。在叙述上以深入浅出的语言结合直观的图示、示例，使读者能够较轻松地理解面向对象编程的基本概念与思想，容易上手，于不知不觉之中掌握 Visual C#编程的方法和技巧。

本书将面向对象的思想贯穿于整个教材，不仅在讲述内容上详细介绍了面向对象的相关概念及编程技巧，而且在所有例题、习题及上机实训中采用“任务驱动”的方式，强调使用面向对象的程序设计方法实现程序功能。强调程序功能由类及其属性、方法等实现，窗体中控件仅组成用户操作界面的“松耦合”程序设计方式（二层架构设计方式）。本书共分为 12 章，主要包括 C#语法基础，流程控制语句与控件，面向对象程序设计方法，数组、结构与集合，接口、委托和事件，泛型，异常处理、程序调试和文件操作，数据库操作，LINQ to SQL 数据库操作等内容。

作者讲授程序设计语言多年并参加过许多实际应用系统的开发，有丰富的教学经验和实践经验。在教材的处理上，以面向对象的程序设计作为主线，以相关的 C#控件作为辅助，通过学习本书，学生不但学会了面向对象程序设计的基本知识、设计思想和方法，还能很容易地过渡到其他语言的学习与使用上。

考虑到初学程序设计的学生往往会出现“上课听得懂，课后不会做”的现象。为方便学生上机练习和教师组织编程训练，还编写了配套辅助教材《C#程序设计教程上机实训与习题解答》对本书中的习题做了详细解答，同时每章还有一个综合性的上机实训。此外，考虑到在 Windows 应用程序开发中报表的重要地位，专门在辅助教材中增加了一章用来介绍使用 C#操作 Excel 文件的相关技术及使用 Excel 生成报表的编程实例。为了便于教学，老师可从机械工业出版社的教材网 http://www.cmpedu.com 下载本书配套教学课件。

本书由刘军、刘瑞新主编，朱立、张治斌、张凡等编著，其中刘军编写第 1、2、12 章，张凡编写第 3 章，刘瑞新编写第 4 章的 4.1～4.5 节，臧顺娟、刘克纯、崔瑛瑛、孙洪玲、徐维维、翟丽娟、缪丽丽、丁新建、骆秋容编写第 4 章的 4.6 节，朱立编写第 5、6 章，李季编写第 7 章，马德龙编写第 8 章的 8.1～8.4 节，岳香菊、彭守旺、庄建新、岳爱英、张国胜、丁新旺、巩义云、李美嫦、胡峰、赵俊杰、李建彬编写第 8 章的 8.5 节，张治斌编写第 9 章的 9.1、9.2 节和第 10 章，王江新编写第 9 章的 9.3、9.4 节和第 11 章，全书由刘瑞新教授统编定稿。

由于作者水平有限，书中疏漏和不足之处难免，敬请广大读者指正。

编 者

教学建议

教学内容	教学要求	课时
第1章 Visual Studio 集成开发环境	了解 Visual Studio 集成开发环境中各窗口的作用及使用方法 了解在 Visual Studio 环境中使用帮助系统的方法 掌握创建简单 Windows 应用程序的基本步骤	2
第2章 C#语法基础	了解 C#变量的命名规范及声明变量的语法 掌握 C#常用数据类型及类型转换（显式转换和隐式转换）方法 掌握 C#常用运算符及表达式 熟练掌握 C#常用数据处理方法（时间日期方法、数学方法、随机方法和字符串方法）	4
第3章 流程控制语句 与控件	熟练掌握顺序结构、选择结构和循环结构程序设计方法 熟练掌握常用控件的属性、事件和方法，能熟练使用流程控制语句配合常用控件设计简单的 Windows 应用程序 理解控件类及动态创建和使用控件的方法 熟练掌握常用键盘鼠标事件的相关概念	8
第4章 面向对象的 程序设计方法	理解面向对象程序设计的概念，理解类、对象的概念及二者之间的关系 熟练掌握创建类文件、类库及声明类成员的方法 理解类的静态成员、方法的重载及参数传递方式 理解类的构造与析构函数及其作用 熟练掌握在应用程序中使用自定义类的方法 了解类的继承与多态性	6
第5章 数组、结构与 集合	熟练掌握一维数组、二维数组的声明、赋值及使用方法 掌握控件数组的声明及使用方法 熟练掌握结构与结构数组的声明、赋值和使用方法 了解枚举的基本概念 熟练掌握 ArrayList 集合与 HashTable 集合的概念、声明、赋值及使用方法，注意对比它们与数组的区别	4
第6章 接口、委托和 事件	理解接口、委托的概念、声明及使用方法 理解事件的概念及创建和使用自定义事件的方法 理解事件与委托之间的关系 了解控件的预定义事件	2
第7章　泛型	了解泛型的概念 掌握泛型的声明及使用方法 掌握 List<T>和 Dictionary<K,V>泛型集合的声明及使用方法 了解泛型方法和泛型接口的概念	2
第8章 异常处理、程序 调试和文件操作	了解 C#应用程序中异常处理的概念及作用 熟练掌握结构化异常处理语句的使用方法 理解应用程序调试相关的常用窗口，能熟练使用“分步执行”和“程序断点”设置 熟练掌握 File 类的常用方法和属性，能使用 File 类成员实现常规文件操作 理解流的概念 熟练掌握 FileStream 的概念及使用方法 熟练掌握对文本文件的读写操作	4

（续）

教 学 内 容	教 学 要 求	课　时
第 9 章 数据绑定 和数据访问控件	理解数据绑定的概念，理解简单绑定和复杂绑定的概念及区别 熟练使用常用数据绑定控件 掌握 BindingSource 和 BindingNavigator 控件的属性设置及使用方法 熟练掌握 DataGridView 控件的外观设置、数据源设置，并能熟练使用 DataGridView 控件显示数据库数据	4
第 10 章 使用 ADO.NET 访问数据库	理解 ADO.NET 的概念及其包含的常用对象 熟练掌握 Connection、Command、DataReader、DataAdapter 及 DataTable 等对象的创建和使用方法，能熟练使用这些对象实现对数据库记录的增、删、改、查等操作	8
第 11 章 使用 DataSet 访问数据库	理解 DataSet 提供的离线数据库操作的概念及工作过程 熟练掌握 DataSet 中常用对象、属性和方法，能熟练使用 DataSet 实现对数据库记录的增、删、改、查等操作	8
第 12 章 LINQ to SQL 数据库操作	理解 LINQ 的基本概念及构成 理解对象初始化器、匿名类型、扩展方法和 Lambda 表达式的概念及使用方法 熟练掌握常用的 LINQ 查询表达式和常用查询方法 掌握 O/R 设计器的使用方法及 DBML 文件的结构 熟练掌握使用 Linq to SQL 实现数据库记录的增、删、改、查等操作	（选学 6）
综合练习	数据库应用系统设计	20（1 周）
总学时		72

说明：

① 建议所有教学内容均在机房进行，采用“边讲边练”、“任务驱动”的教学模式。

② 建议每次课后选择习题中的部分题目作为课外作业；每章结束后，以课外作业的形式安排学生完成本书配套教材《C#程序设计教程上机实训与习题解答》中的实训项目。任课教师可根据学生完成情况给予讲评，对普遍出现的问题给予集中解答。

③ 课程结束时安排 1 周的“综合练习”，以巩固所学知识，提高学生分析问题和解决问题的能力。

④ 本课程建议教学时数为 72～78 学时（包括 1 周综合练习），任课教师可根据实际情况酌情增减或有选择地讲解部分章节的内容。

目　录

第 1 章　Visual Studio 集成开发环境

Visual Studio 是美国微软公司推出的一套完整的应用程序开发工具集。在 Visual Studio 所支持的众多程序设计语言中，C#和 Visual Basic 是最主要的两种。本书以 Visual Studio 2008 为背景介绍使用 C#语言进行 Windows 应用程序设计的基本技术，但所述概念及演练、习题等除特别声明外均可兼容 Visual Studio 2005 甚至更早的版本。

Visual Studio 2008 内置.NET Framework 版本为 3.5，同时提供对.NET Framework 2.0、3.0 的支持。Visual Studio 可用于 Windows 应用程序、ASP.NET Web 应用程序、XML Web Services 和移动应用程序的开发，它将程序设计中需要的各个环节（界面设计、程序设计、运行、调试和部署应用程序等）集成在同一个窗口中，极大地方便了开发人员的设计工作。通常将这种集多种功能于一身的开发平台称为集成开发环境（IDE）。

1.1　Visual Studio 的项目管理

在 Visual Studio 开发环境中，所有的程序均属于一个“项目”（Windows 应用程序）或“网站”（ASP.NET 应用程序），其中通常包括了窗体文件、类文件和其他一些与程序相关的文件，所以项目管理是在 Visual Studio 集成开发环境中进行 Windows 应用程序设计工作的基础。

1.1.1　Visual Studio 的初始设置

初次运行 Visual Studio 时系统将显示图 1-1 所示的“选择默认环境设置”对话框，要求进行开发环境的初始配置。选择不同的项目会导致 Visual Studio 的菜单命令、工具箱等具有不同的内容。使用 C#语言进行 Windows 应用程序开发，应在“选择默认环境设置”列表中选择“Visual C#开发设置”后单击“启动 Visual Studio”按钮进入开发环境。

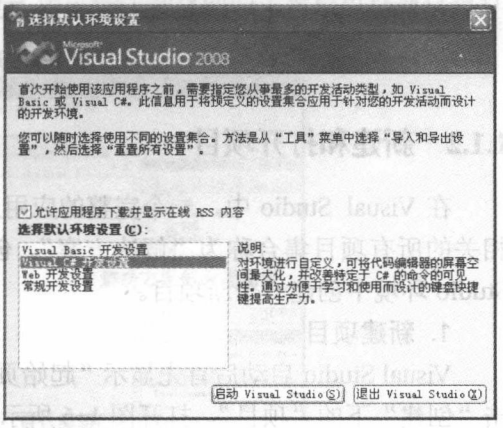

图1-1　“选择默认环境设置”对话框

如果在使用的过程中想改变默认环境设置，可以在 Visual Studio 主界面中执行“工具”菜单下的“导入和导出设置”命令，通过图 1-2 所示的“导入和导出设置向导”对话框对当前设置进行必要的修改。在对话框中，选择“重置所有设置”单选按钮，单击“下一步”按钮，在图 1-3 所示的“保存当前设置”对话框中选择“否，仅重置设置，从而覆盖我的当前设置”单选按钮，单击“下一步”按钮。如果选择“是，保存我的当前设置”单选按钮，并指定文件名称和保存位置，可将当前配置保存在指定文件中，以备将来再次使用时导入。

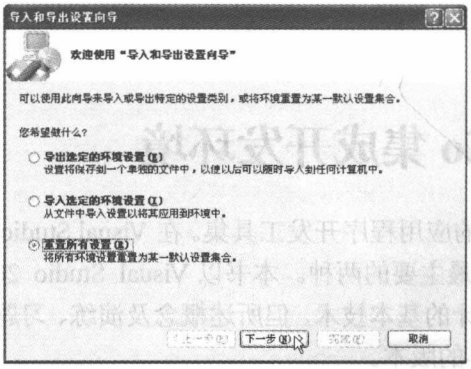

图1-2 “导入和导出设置向导”对话框

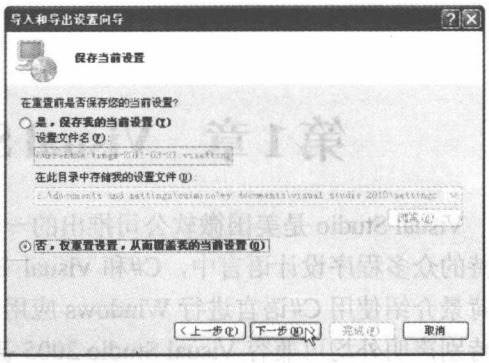

图1-3 是否保存当前设置

在图 1-4 所示的对话框中，选择新的环境设置（如 Web 开发），单击“完成”按钮。

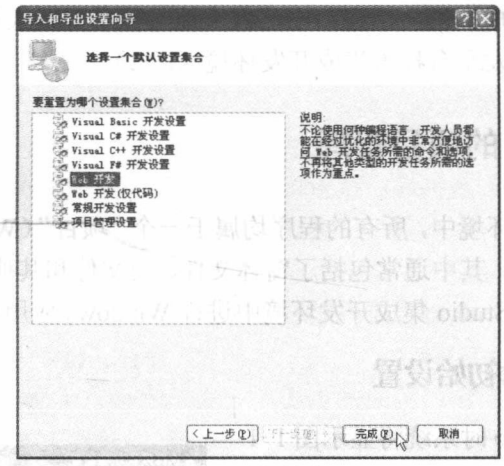

图1-4 选择新的开发环境

1.1.2 新建和打开项目

在 Visual Studio 中，一个完整的应用系统可能会包含在若干个“项目”中，与应用系统相关的所有项目集合称为“解决方案”。创建一个 Windows 应用程序，首先需要在 Visual Studio 环境中创建一个新项目。

1. 新建项目

Visual Studio 启动后首先显示“起始页”的内容，如图 1-5 所示在“最近的项目”栏中单击“创建”下的“项目”，打开图 1-6 所示的“新建项目”对话框，选择“Visual C#”模板下的“Windows 窗体应用程序”，选择使用的.NET Framework 版本（默认为.NET 3.5，同时提供对.NET 2.0、3.0 的支持），并指定项目名称、保存位置、解决方案名称后单击“确定”按钮，系统将根据用户设置自动创建一个“空白”的 Windows 应用程序框架。

2. 打开现有项目

如果希望打开已创建但尚未完成所有设计工作的项目，可以通过单击“起始页”中“打开”下的“项目”，在打开的对话框中选择项目文件（*.sln）将其打开。也可以通过 Windows

“我的电脑”或“资源管理器”直接双击保存在项目文件夹中的*.sln 文件并将其打开。对于最近使用过的项目会出现在“最近的项目”窗格中，单击某项目名称可将其再次打开，进入到 Visual Studio 集成开发环境中。

图1-5　创建新项目

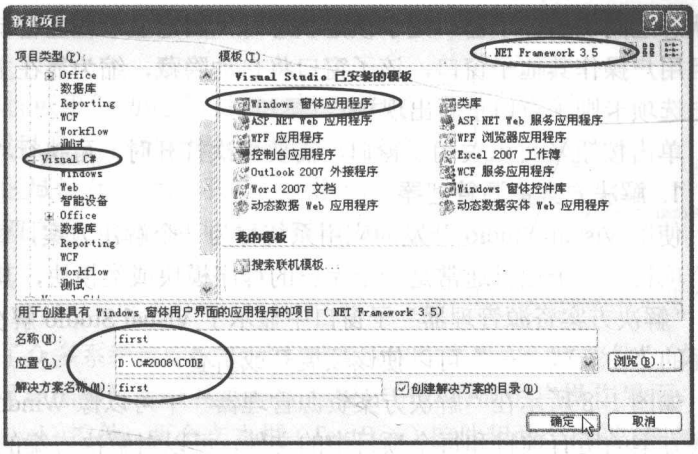

图1-6　选择模板和.NET Framework版本

1.1.3　集成开发环境中的主要子窗口

创建了一个新项目后，系统进入图 1-7 所示的 Visual Studio 集成开发环境主界面。从图中可以看到主界面中除了具有菜单栏、工具栏外，还包含了许多子窗口，其中最常用的是“工具箱”、“解决方案资源管理器”和“属性”子窗口。用户的主要工作区域是“窗体编辑区”，该子窗口用来显示 Windows 窗体的“设计”视图（“Form1.cs[设计]”选项卡）或程序的“代码窗口”（“Form1.cs”选项卡）。更多用于管理项目、调试程序等目的的子窗口可以通过执行“视图”菜单中的相应命令将其打开。

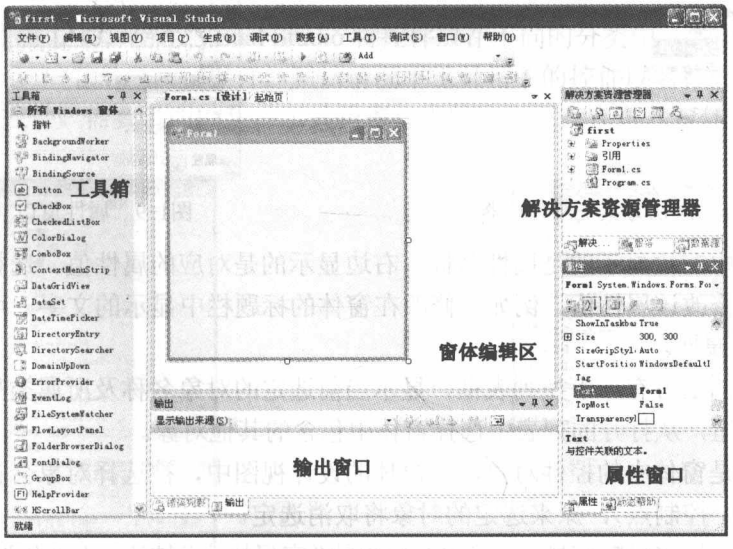

图1-7　Visual Studio集成开发环境主界面

在大多数子窗口的右上角都有 ▾ 📌 ✕ 三个按钮。单击按钮 ▾ 将弹出一个关于子窗口管理的菜单，包括“浮动”、“可停靠”、“选项卡式文档”、“自动隐藏”、“隐藏”等命令。

“自动隐藏”按钮 📌 是一个开关按钮，单击该按钮时可在 📌（锁定）或 ⊣（自动隐藏）两种状态之间切换。子窗口处于锁定状态时，固定显示在主窗口中；处于“自动隐藏”状态时一旦用户操作其他子窗口，该子窗口将自动隐藏，缩为靠在主窗口边框上的一个选项卡，若单击选项卡则子窗口重新出现。

单击按钮 ✕ 可以关闭子窗口，需要再次打开时，可执行“视图”菜单中的相应命令。

1. 解决方案资源管理器

使用 Visual Studio 开发的应用系统称为一个解决方案，每一个解决方案可以包含一个或多个项目。一个项目通常是一个完整的程序模块或类模块，其中可能包含有若干文件。

“解决方案资源管理器”子窗口中显示了 Visual Studio 解决方案的树型结构，单击项名称前面的“＋”或“－”可以使该项展开或折叠。

如图 1-8 所示在“解决方案资源管理器”中可以像 Windows 资源管理器那样，浏览组成解决方案的所有项目和每个项目中的文件，可以对解决方案的各元素进行操作，如打开文件、添加内容、重命名、删除等。在解决方案资源管理器中，双击某个文件，将在主窗口中显示相应的视图（设计视图或代码窗口），以便对该文件进行编辑。

2. 属性窗口

“属性”子窗口用于设置解决方案中各对象的属性，当选择 Windows 窗体的设计视图、项目名称或类视图中的某一项时，属性子窗口将以两列表格的形式显示该子项的所有属性，图 1-9 所示的是在 Windows 窗体中选择了窗体控件 Form1 时显示的属性内容。

图1-8　解决方案资源管理器

图1-9　属性窗口

在属性窗口中左边显示的是属性名称，右边显示的是对应的属性值。选择某一属性名称后，可以在右边修改该属性值。例如，修改在窗体的标题栏中显示的文字，可直接在属性窗口中修改其 Text 属性。

在属性窗口的上方有一下拉列表框，显示当前选定的对象名称及所属类型。可以单击该列表框的下拉按钮，从打开的列表中选择窗体中包含的其他对象。

如果选择的是窗体中的控件对象，在窗体的设计视图中，被选择对象会自动处于选定状态（四周出现八个控制点），原来选定的对象将取消选定。

属性名称默认按字母顺序排列，单击窗口中的“字母排序”按钮 🔤 与“分类排序”按钮 ⊞，可以在两种排序方式之间切换。

选择设计视图中的窗体或窗体中的某控件，并在属性窗口中单击“事件”按钮⚡，属性子窗口中将显示被选择窗体或控件支持的所有事件列表，如图 1-10 所示。双击某一事件名称，将自动打开代码窗口，并添加该事件处理程序的声明，用户可在其中输入处理响应事件的程序代码。

图1-10　文本框控件的事件列表

属性窗口的最下方有一个属性或事件功能的说明区域，当选择某一属性或事件时，说明区域显示文字说明属性或事件的作用，这对初学者很有用。如果该区域没有显示，可将鼠标指向窗口列表框下部边框，当鼠标变为双向箭头时，向上拖动鼠标，该区域即可显示出来。

3. 工具箱

Visual Studio 工具箱中存放了众多系统预定义的用于组成应用程序界面的“控件”（如，文本框、标签、按钮等），用户可根据实际需要将其添加到窗体中以构成用户操作界面，图 1-11 所示的就是由两个标签（Label）控件、两个文本框（TextBox）控件和两个按钮（Button）控件构成的登录对话框界面。

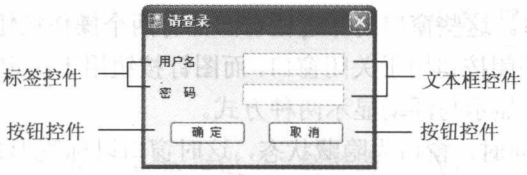

图1-11　由窗体和各类控件构成的程序界面

默认状态下，工具箱处于“自动隐藏”状态，窗口的左边框处有工具箱的选项卡标签。当鼠标指向该标签时，工具箱将显示到屏幕中。

工具箱用于向应用程序窗体中添加控件以构成应用程序的基本界面。利用工具箱使用户可以像“拼图”一样简单地组成程序界面。

如图 1-12 和图 1-13 所示，Visual Studio 将控件放在不同的分类中，各分类卡以“+”号表示折叠状态，以“-”号表示已展开。默认情况下工具箱中的控件以名称的字母顺序排列，方便用户的查找控件。需要说明的是，工具箱窗口中的内容只有在进入窗体设计视图时才会显示出来。

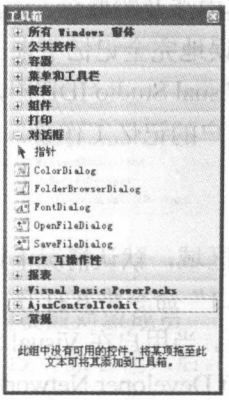

图1-12　工具箱中控件的分类

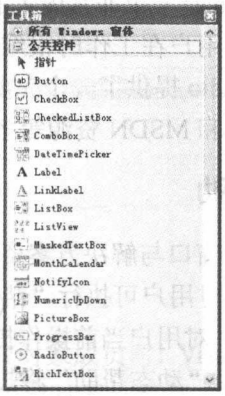

图1-13　公共控件

如果进入了窗体的设计视图，工具箱窗口仍没有出现在 Visual Studio 的 IDE 环境中，可执行“视图”菜单中的“工具箱”命令将其打开。

用户不但可以从工具箱中选择控件并将其拖动到窗体中，还可以将某一代码片段拖动到工具箱中暂存，以便将来重新使用。例如，可以将按钮（Button）控件从工具箱中拖放到窗体设计视图中，即向窗体中添加控件。也可从代码窗口中选择并拖动一个代码片段到工具箱中，将来需要重复使用该代码片段时，可将其拖回代码窗口的适当位置。

4. 子窗口的操作

在 Visual Studio 集成开发环境中，有两类子窗口，一类是在窗体编辑区显示的窗口，如“代码窗口”、“设计”视图窗口等；另一类是在窗体编辑器周围显示的子窗口，如“工具箱”、“解决方案资源管理器”、“服务器资源管理器”、“类视图”、“属性”、“输出”等。

如果在窗体编辑区显示的窗口不止一个，则诸多窗口将以选项卡的形式叠放在一起，在最前端显示的为当前活动窗口，用户可以通过单击选项卡的标签实现各窗口的切换。在窗口的右上角有一个关闭按钮×，用于关闭窗口，关闭按钮只对当前活动窗口有效。要关闭某一窗口，首先要使该窗口成为当前活动窗口，然后单击关闭按钮。

在窗体编辑区周围的窗口也是由若干子窗口共享某一屏幕区域，以选项卡的形式叠放在一起，通过标签切换窗口。这些窗口的标题栏右部都有两个操作按钮，一个是关闭按钮×，另一个是图钉按钮 。关闭按钮用于关闭窗口，而图钉按钮用于决定窗口的隐藏与显示状态，在显示状态又可以有停靠显示与浮动显示两种方式。

当图钉按钮 为横向时，窗口为隐藏状态，这时窗口以标签形式显示在 Visual Studio 的左、右、下边框，这时用鼠标指向标签，窗口方显示，鼠标移出窗口，则窗口又重新隐藏。隐藏状态的主要作用是为主区域的窗口开辟尽可能大的屏幕空间。

当图钉按钮 为纵向时，窗口为显示状态，默认为停靠方式，即窗口附着在 Visual Studio 的左、右、下边框。这时鼠标指向窗口的标题栏，拖动鼠标，使窗口离开边框，窗口即为浮动显示方式，这时标题栏上的图钉按钮将消失。如要使浮动方式变为停靠方式，只需拖动窗口至 Visual Studio 主窗口的边框上即可。

1.2 Visual Studio 的帮助系统

Visual Studio 是一个非常庞大的应用程序开发系统，其中涉及的概念、语法、函数自然是很多的，这使得用户在工作时很难将需要的知识点准确无误地完全记忆下来。为了解决这一问题，Visual Studio 提供了一个完备的帮助系统，用户在 Visual Studio IDE 中，可以使用动态帮助、智能感知和 MSDN 资源等多种帮助形式，减少了用户的记忆工作量。

1.2.1 动态帮助

动态帮助子窗口与解决方案资源管理器共享一个屏幕区域，默认情况下该窗口并未显示到屏幕上，需要时用户可执行“帮助”菜单中的“动态窗口”命令将其打开。

“动态帮助”对用户当前操作提供相关的帮助主题列表。当用户在 Visual Studio 环境中进行某一项操作时，“动态帮助”将搜索 MSDN 库（Microsoft Developer Network），查找与该操作相关的帮助主题，以超级链接显示在“动态帮助”窗口，并把它认为可能最有用的主题列

在第一位。单击一个主题链接后，此主题的内容将会显示在 Visual Studio 的帮助窗口中。

例如，当在设计器窗口中选择一个按钮控件时，“动态帮助”显示的帮助主题内容如图 1-14 所示；当在代码窗口中选择关键字 new 时，“动态帮助”窗口会显示已选定的关键字 new 的帮助主题，如图 1-15 所示。

图1-14　选择按钮控件后的动态帮助

图1-15　选择关键字new后的动态帮助

1.2.2　智能感知

在代码编写的过程中，Visual Studio 提供了“智能感知”的帮助方式，利用这种帮助方式不仅可以节省代码输入的时间，更重要的是避免了用户的输入错误。Visual C#提供了提示类名或对象名、提示类成员或对象成员、提示方法的使用说明等。

1. 提示类名或对象名

在设计代码的过程中，当输入类名或对象名时，Visual Studio 会动态提供当前可用的类及对象列表，如果选择某一列表项，则动态显示该项简要说明，如图 1-16 所示。用户可以通过鼠标双击所选择项完成自动输入，也可以用键盘的上、下方向键选择所需项然后按〈Enter〉键，如果在输入的过程中自动选择了所需项，则可以直接按〈Enter〉键。这样可以避免输入错误，特别是大小写错误。

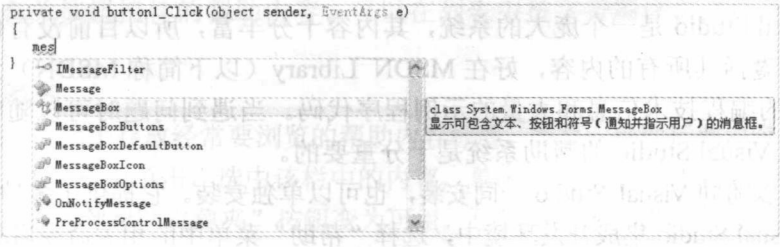

图1-16　提示类名或对象名

2. 提示类成员或对象成员

在完成类名或对象名输入后，当通过点运算符访问类或对象成员时，Visual Studio 将动态显示类成员或对象成员列表框，可以通过鼠标或键盘选择这些成员。例如，当完成整型变量