



高职高专“十一五”规划教材

计算机类

C#程序设计 项目化教程

徐超 翟鹏翔 主编



冶金工业出版社

www.cnmp.com.cn

高职高专“十一五”规划教材·计算机类

C#程序设计项目化教程

主 编 徐 超 翟鹏翔
副主编 黄 菊 陈 彦
主 审 葛红美

北 京
冶金工业出版社

2009

内 容 简 介

本书以开发人力资源管理系统为主线,介绍了 C#在软件项目开发时的具体应用。全书共分为 4 个软件开发情景,主要内容有:产品安全登录验证、产品界面设计、产品数据管理、产品发布。

本书基于工作过程导向出发,以学生为主体,重点解决学生分析问题、解决问题的能力,设置了 4 个情景教学,每一情景下有不同的项目任务,使每一模块知识化整为零,在课程最后利用综合实训将所有任务知识串起来,即人力资源管理系统(MIS)。

本书实用性强,适合作为高职高专院校计算机及相关专业的教材,也适合作为程序员的自学参考书。

图书在版编目(CIP)数据

C#程序设计项目化教程/徐超,翟鹏翔主编. —北京:冶金工业出版社, 2009.8

ISBN 978-7-5024-4539-3

I. C… II. ①徐…②翟… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 153757 号

出 版 人 曹胜利

地 址 北京北河沿大街嵩祝院北巷 39 号, 邮编 100009

电 话 (010)64027926 电子信箱 postmaster@cnmip.com.cn

责任编辑 刘 源

ISBN 978-7-5024-4539-3

北京天正元印务有限公司印刷; 冶金工业出版社发行; 各地新华书店经销

2009 年 8 月第 1 版, 2009 年 8 月第 1 次印刷

787mm×1092mm 1/16; 12 印张; 281 千字; 185 页; 1-3000 册

26.00 元

(本书如有印装质量问题, 本社发行部负责退换)

前 言

C#是一种安全的、稳定的、简单的,由C和C++衍生出来的面向对象的编程语言。它在继承C和C++强大功能的同时去掉了一些它们的复杂特性。C#综合了VB简单的可视化操作和C++的高运行效率,以其强大的操作能力、优雅的语法风格、创新的语言特性和便捷的面向组件编程的支持成为.NET开发的首选语言。

本书基于工作过程导向出发,以目前最典型的管理系统——人力资源管理系统贯穿全书,要求每章每个模块中要体现出如何开发该系统的过程,最后进行综合项目实训。学生学习本书后,可以独立完成的开发出人力资源管理系统,这是本书一大特色。

本书的编写思想如下:

(1) 本书采用情景教学法,设置了4个情景教学,每一情景下有不同的项目任务,使每一模块知识化整为零,在课程最后利用综合实训将所有任务知识串起来。

(2) 以学生为主体,以学生就业技能为出发点与落脚点,力求让学生用最简单的方法、最少的时间学到最有用的系统开发技术(就业)为目的。

(3) 遵循从实际到理论、从具体到抽象、从个别到一般的人类认识客观事物的方法;提出问题,介绍解决问题的方法,归纳规律和概念。

(4) 着重“怎么做?”,而不去纠缠“为什么?”,着眼点是应用,而不是系统地讲解理论。

(5) 力求简明扼要,深入浅出,通俗易懂,看得懂、学得会。

本书是对计算机职业教育教学改革的一种尝试,全书由徐超、翟鹏翔任主编,黄菊、陈彦任副主编,王志强、李佼辉、王丽艳、谷广超、康丽军、赵娟、王晶晶参加编写,全书由葛红美统稿。书中所有代码均在Visual Studio 2005中编译调试成功。

由于编者水平所限,书中如有不足之处敬请使用本书的师生与读者批评指正,以便修订时改进。如读者在使用本书的过程中有其他意见或建议,恳请向编者(bjzhangxf@126.com)踊跃提出宝贵意见。

编 者

目 录

情景一 产品安全登录验证..... 1

一、知识目标.....	1
二、能力目标.....	1
三、工作任务.....	1
四、预备知识.....	1
模块一 我的 QQ 登录.....	4
一、知识目标.....	4
二、工作任务.....	4
三、操作流程.....	4
四、知识拓展.....	11
模块二 QQ 登录验证.....	12
一、知识目标.....	12
二、工作任务.....	12
三、操作流程.....	12
四、知识拓展.....	17

情景二 产品界面设计..... 18

一、知识目标.....	18
二、能力目标.....	18
三、工作任务.....	18
四、预备知识.....	18
模块一 系统登录窗体以及主窗体 的设计.....	21
一、知识目标.....	21
二、工作任务.....	22
三、操作流程.....	22
四、知识拓展.....	38
模块二 信息录入和查询界面.....	40
一、知识目标.....	40
二、工作任务.....	40
三、操作流程.....	40
四、知识拓展.....	63
模块三 多媒体技术.....	67
一、知识目标.....	67

二、工作任务.....	67
-------------	----

三、操作流程.....	67
-------------	----

四、知识拓展.....	81
-------------	----

情景三 产品数据管理..... 85

一、知识目标.....	85
二、能力目标.....	85
三、工作任务.....	85
四、预备知识.....	86
模块一 ADO.NET 连接数据库.....	91
一、知识目标.....	91
二、工作任务.....	91
三、操作流程.....	91
四、知识拓展.....	102
模块二 用户登录模块.....	102
一、知识目标.....	102
二、工作任务.....	103
三、操作流程.....	103
四、知识拓展.....	112
模块三 考生查询模块.....	112
一、知识目标.....	112
二、工作任务.....	112
三、操作流程.....	112
四、知识拓展.....	121
模块四 考生信息添加模块.....	122
一、知识目标.....	122
二、工作任务.....	122
三、操作流程.....	122
四、知识拓展.....	128
模块五 考生信息浏览模块.....	128
一、知识目标.....	128
二、工作任务.....	128
三、操作流程.....	129
四、知识拓展.....	140

模块六 当前用户密码修改模块	140	三、操作流程	147
一、知识目标	140	四、知识拓展	158
二、工作任务	140	模块二 帮助文件	159
三、操作流程	141	一、知识目标	159
四、知识拓展	143	二、工作任务	159
情景四 产品发布	144	三、操作流程	159
一、知识目标	144	四、知识拓展	165
二、能力目标	144	模块三 系统发布	171
三、工作任务	144	一、知识目标	171
四、预备知识	144	二、工作任务	171
模块一 水晶报表	146	三、操作流程	171
一、知识目标	146	四、知识拓展	180
二、工作任务	146	参考文献	185

情景一 产品安全登录验证

一、知识目标

(一)项目目标

能够熟悉 C#开发环境，利用 C#工具实现产品安全登录验证功能。

(二)教学目标

- (1) 掌握 C#开发工具的安装。
- (2) 熟悉 C#开发工具环境，了解 C#开发工具的组成。
- (3) 掌握 C#工程项目的新建、保存与退出。
- (4) 熟悉 C#文件类型。
- (5) 熟悉 C#常用组件的用法，并能设计出用户登录窗体。
- (6) 掌握 C#语言的应用。

二、能力目标

- (1) 培养学生收集、处理信息，以及准备和动手的能力。
- (2) 理解 C#桌面开发的理念。
- (3) 能够进行友好界面的设计。
- (4) 具备用 C#语言编写程序的能力。
- (5) 培养学生具备产品安全登录验证设计开发能力。

三、工作任务

分别使用 C#工具中的 Label、ComboBox、Button 等组件，制作软件产品中的安全登录验证窗体。

四、预备知识

(一)进入 C#世界

说起 C#就不能不提到.NET Framework。 .NET Framework(框架)简称.NET，是微软为开发应用程序创建的一个富有革命性的新平台。自 2002 年发布这一版以来，它已经迅速占领企业应用市场，全球财富 100 强的公司有 90%用.NET 技术构建信息系统，再次演绎了微软不败的神话。在这个平台环境中，可以开发出运行在 Windows 上的几乎所有应用程序，而微软也将推出运行在其他操作系统上的版本。简单地说，.NET Framework 是一个创建、部署和运行应用程序的多语言多平台环境，包含了一个庞大的代码库，各种.NET 语言都可以

共用这些代码库。

C#(读作: C Sharp)是一种强大的、面向对象的程序开发语言,是专门用于.NET 的编程语言,用 C#编写的所有代码总是在.NET Framework 中运行。使用 C#可以编写出传统的 Windows 桌面应用程序(WinForms)、Windows 服务程序(Windows Service)、Internet 应用程序(ASP.NET)和 Web 服务程序(Web Service)。因此,在当前的软件开发行业中,C#已经成为了绝对的主流语言,可以说 C#语言和 Java 语言在当今企业应用中已经平分天下。

微软推出了这么强大的平台和技术,当然也会有强大的集成开发环境(IDE)来支持,那就是微软提供的 Visual Studio .NET(VS)。它集成了.NET Framework,利用 Visual Studio .NET,可以快速开发.NET 应用程序。

随着.NET Framework 的不断升级,微软也在不断升级 Visual Studio,本书使用的是当前(教材编写时)最新的版本 Visual Studio 2005,它支持的是.NET Framework 2.0。VS 2005 开发环境如图 1-1 所示。

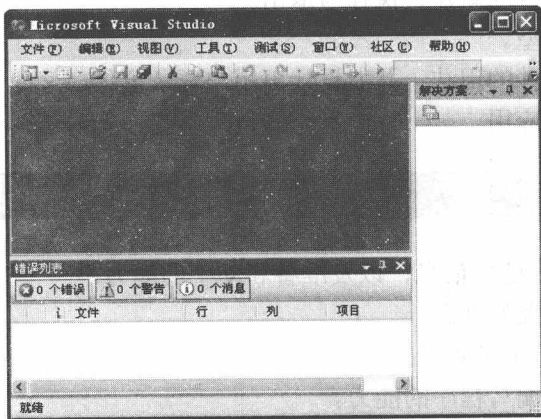


图 1-1 VS 2005 开发环境

(二)C#语言基础

C#是微软公司发布的一种面向对象的、运行于.NET Framework 之上的高级程序设计语言。

微软 C#语言定义主要是从 C 和 C++继承而来的,而且语言中的许多元素也反映了这一点。C#设计者从 C++继承的可选项方面比 Java 要广泛一些(例如 structs),它还增加了自己新的特点(例如源代码版本定义),但它还太不成熟,不可能挤垮 Java。C#还需要进化成一种开发者能够接受和采用的语言,而微软当前为它的这种新语言大造声势也是值得注意的。目前大家的反应是:“这是对 Java 的反击。”

C#是完全面向对象的,在 C#里面只有类,没有所谓的全局变量。C#里面取消了指针的使用,还有就是在 C#里面一个实例的声明必须使用 new(如: Car car=new Car()此处的 new 表示一个操作符),它抛弃了多继承、拥有自动内存管理(垃圾收集),是.NET 平台的首选语言。

(三)第一个 C#程序

对.NET Framework、C#和 VS 有了初步的认识后,下面来看一看 C#程序的庐山真面目。

首先启动 VS, 选择“开始”|“程序”| Microsoft Visual Studio 2005 | Microsoft Visual Studio 2005 选项, 启动 Microsoft Visual Studio 2005, 如图 1-2 所示。



图 1-2 启动 Microsoft Visual Studio 2005

使用 VS 2005 创建并运行 Windows 应用程序包括 3 步: 新建项目、生成解决方案、调试。现在就创建第一个 Windows 应用程序, 在屏幕中显示“欢迎进入 VS2005!”。

1. 新建项目(项目: project)

- (1) 在 VS 菜单栏中选择“文件”|“新建”|“项目”命令, 打开“新建项目”对话框。
- (2) 在左侧的项目类型中选择 Visual C#选项, 在右侧的模版列表中选择“Windows 应用程序”选项。
- (3) 在“名称”文本框中输入 Windows Application1。
- (4) 为项目选择一个保存的位置, 例如 D:\a1, 单击“确定”按钮后, 就创建了一个 C# 代码模版, 如图 1-3 所示。

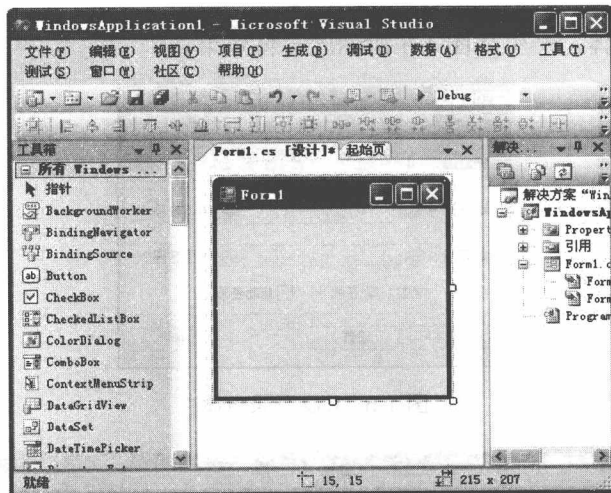


图 1-3 C#代码模版

在 Form1 窗体中添加 Label1 组件, 在该组件属性 Text 中输入“欢迎进入 VS2005!”。

2. 生成解决方案(生成: build, 解决方案: solution)

在 VS 菜单栏中选择“生成”|“生成解决方案”命令。如果错误列表中没有显示错误和警告, VS 的状态区中显示“生成成功”, 就表示代码没有编译错误, 可以运行了。

3. 调试(调试: debug)

在 VS 菜单栏中选择“调试”|“启动调试”命令。如果看到如图 1-4 所示的结果, 恭喜你已经成功运行出自己的第一个 C#程序了!

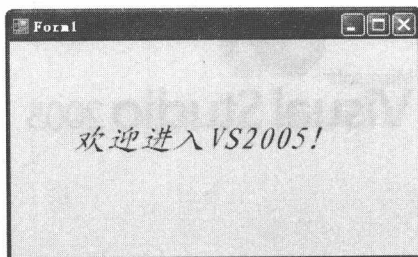


图 1-4 Windows Application1 项目输出

模块一 我的 QQ 登录

一、知识目标

- (1) 熟悉 C#基本环境。
- (2) 了解 C#编程代码规范。
- (3) 掌握 C#开发工具的基本使用: 新建工程项目、保存与退出、组件的基本用法。

二、工作任务

制作 QQ 登录窗体, 产生如图 1-5 所示的窗体效果。

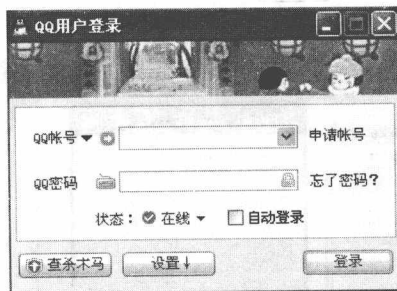


图 1-5 QQ 登录窗体

三、操作流程

- (1) 新建工程项目。选择“开始”|“程序”| Microsoft Visual Studio 2005 | Microsoft Visual Studio 2005 选项, 启动 Microsoft Visual Studio 2005。在打开的窗口中选择“文件”|“新建”

1 “项目” | “Windows 应用程序” 命令。

(2) 在 Form 窗体中添加组件。在 Form 窗体中加入 1 个 PictureBox 组件、1 个 GroupBox 组件、5 个 Label 组件、1 个 ComboBox 组件、1 个 CheckBox 组件和 3 个 Button 组件。产生的效果如图 1-6 所示。



图 1-6 设计窗体效果

(3) 设置各组件属性，见表 1-1。

表 1-1 属性设置表

组件面板	所用组件	组件属性	属性设为
Form 窗体	—	Text	QQ 用户登录
Form 窗体	—	Icon	添加 QQ 图标
Form 窗体	—	MaximizeBox	False
Form 窗体	—	AutoSizeMode	GrowAndShrink
公共控件	PictureBox	选择图像	添加 QQ 图片
公共控件	Label1	Text	QQ 账号:
公共控件	Label2	Text	QQ 密码:
公共控件	Label3	Text	状态:
公共控件	Label4	Text	申请账号
公共控件	Label5	Text	忘了密码?
容器	GroupBox	Text	—
公共控件	ComBoBox	—	—
公共控件	TextBox	—	—
公共控件	CheckBox	Text	自动登录
公共控件	Button1	Text	查杀木马
公共控件	Button2	Text	设置
公共控件	Button2	Text	登录

(4) 设置窗体界面。设置好各个属性后，根据如图 1-6 所示的效果进行界面设计。在窗体中右击 Position 控件可以调整组件的位置；指向某个组件时，右击 Control 控件可以调整组件的重叠。

(5) 保存工程项目。选择“文件” | “保存”命令即可保存工程项目。

(6) 运行工程项目。选择“调试” | “启动调试”命令，或者单击工具栏中的图标，即可运行工程项目，查看结果。



知识要点



● C#的安装

将C#的安装光盘放入光驱中,系统会自动运行安装程序,显示安装画面;或者双击桌面上“我的电脑”图标,打开光盘双击 Setup.exe 图标,将出现安装画面。根据安装向导提示或者一路单击“下一步”按钮,即可安装成功。

● C#开发环境

C#的主窗体放置着大部分程序管理和开发工具,最上方为菜单栏,右侧为工具箱,中间为窗体,左侧为解决方案资源管理器,最下方为编译、调试错误列表。

(1) 菜单栏。菜单栏集成了大部分命令(如图 1-7 所示),在下面会详细阐述。

(2) 工具栏。工具栏提供了快速打开、关闭、添加、删除和保存等按钮,包含了菜单栏中的大部分常用命令。

(3) 工具箱。工具箱按类放置各种控件,是C#编程的核心。

(4) 解决方案资源管理器。解决方案资源管理器用于定制集成开发环境。

● 菜单栏

菜单栏在标题栏的下面,包含C#的各项命令,如图 1-7 所示。单击菜单栏中的任意一个菜单便会展开下拉式菜单。每个下拉式菜单均有一组功能相近的命令可供选择,命令为灰色时不可选,对不同的操作环境和对象,菜单的颜色不同。通过单击这些菜单中的命令,就能够实现C#提供的某个功能。例如要打开一个文件,只要将鼠标移动到“文件”菜单上,然后单击,这时会弹出一个菜单。在此菜单里有“打开”命令,选择该命令即可。菜单栏包括以下内容:

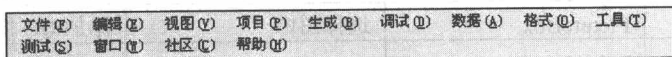


图 1-7 菜单栏

(1) “文件”菜单。“文件”菜单包括“新建”、“打开”、“保存 Form1.cs”、“打印”等命令,如图 1-8 所示。

(2) “编辑”菜单。“编辑”菜单提供了各种编辑功能,如“删除”、“复制”和“粘贴”等命令,如图 1-9 所示。



图 1-8 “文件”菜单

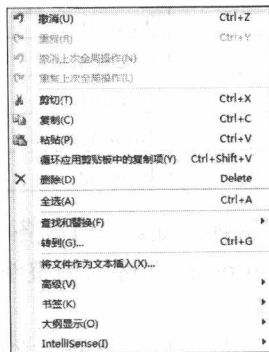


图 1-9 “编辑”菜单

(3) “视图”菜单。“视图”菜单用于打开 C#环境各个窗口和项目中的窗体和单元文件等，如图 1-10 所示。

(4) “项目”菜单。“项目”菜单用于添加处理 C#中的各种类型的组件，如图 1-11 所示。

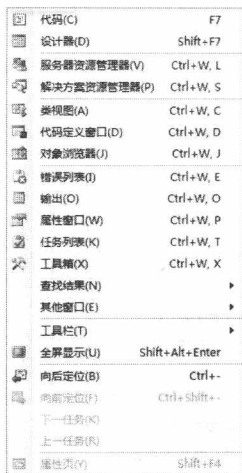


图 1-10 “视图”菜单

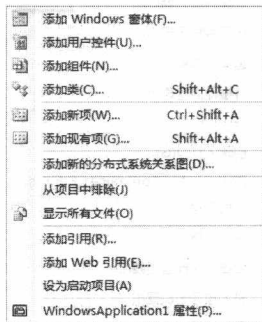


图 1-11 “项目”菜单

(5) “生成”菜单。“生成”菜单包括“生成解决方案”、“重新生成解决方案”等命令，如图 1-12 所示。

(6) “调试”菜单。“调试”菜单用于运行、调试应用程序，单步执行程序等，如图 1-13 所示。



图 1-12 “生成”菜单



图 1-13 “调试”菜单

(7) “数据”菜单。“数据”菜单用于对数据源的操作，如图 1-14 所示。

(8) “格式”菜单。“格式”菜单用于对控件摆放位置的调整操作，如图 1-15 所示。

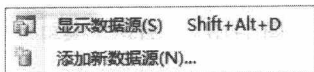


图 1-14 “数据”菜单

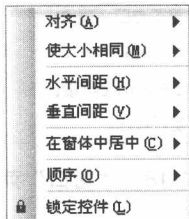


图 1-15 “格式”菜单

(9) “工具”菜单。“工具”菜单包括“附加到进程”、“连接到设备”等命令，如图 1-16 所示。

(10) “测试”菜单。“测试”菜单包括“新建测试”、“加载元数据文件”、“创建新测试列表”等命令，如图 1-17 所示。

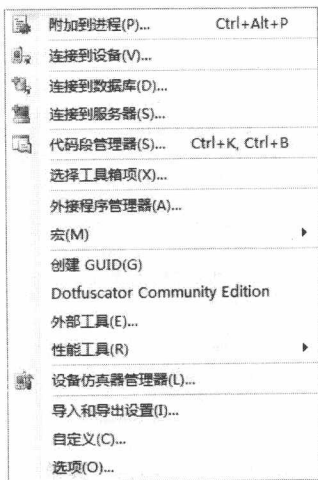


图 1-16 “工具”菜单

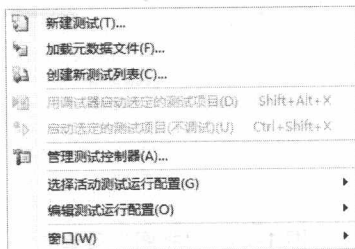


图 1-17 “测试”菜单

(11) “窗口”菜单。“窗口”菜单提供打开可视化环境各个窗口的命令，如图 1-18 所示。

(12) “社区”菜单。“社区”菜单包括“发送反馈意见”、“社区搜索”等命令，如图 1-19 所示。

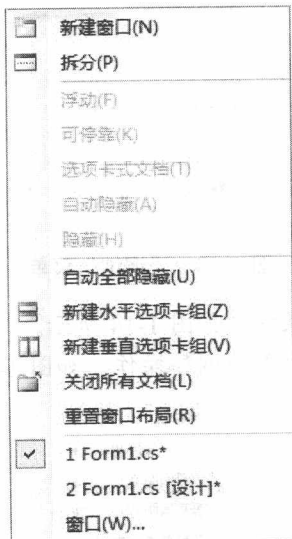


图 1-18 “窗口”菜单

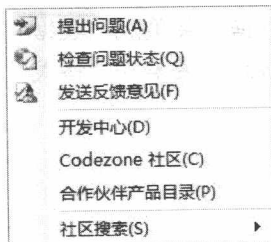


图 1-19 “社区”菜单

(13) “帮助”菜单。“帮助”菜单提供 C#帮助、工具帮助以及在线帮助等功能，如图 1-20 所示。

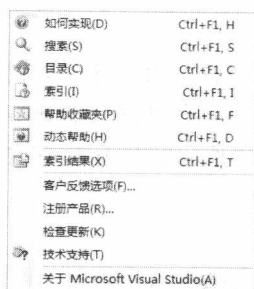


图 1-20 “帮助”菜单

● 工具栏

工具栏中的按钮执行着菜单栏中的一些常用命令，但操作起来更方便，所以又称为快速按钮。当鼠标指向某一按钮，停留大约 2s 后，便会出现提示信息。工具栏如图 1-21 所示。



图 1-21 工具栏

工具栏各按钮功能依次为：新建项目、添加新项、打开文件、保存、全部保存、剪切、复制、粘贴、撤销、重复、向前定位、向后定位、调试、解决方案、解决方案平台在文件中查找、解决方案资源管理器、属性窗口、对象浏览器、工具箱、起始页、命令窗口、对齐到网络、左对齐、居中对齐、右对齐、顶部对齐、中部对齐、底部对齐、宽度相同、高度相同和大小相同等。

● 解决方案资源管理器

C#采用面向对象的程序设计方法，因而对象显得尤为重要。解决方案资源管理器是C#用来控制对象的一个工具，利用它可以方便的查看和修改对象属性，控制和编写对象响应的事件。解决方案资源管理器由标题栏、对象选择下拉列表框以及属性选项卡和事件选项卡组成，如图 1-22 所示。

● 属性设置栏

属性设置栏用来设置 Form 窗体各组件的属性，如字体、颜色、大小、连接及载入等功能，如图 1-23 所示。

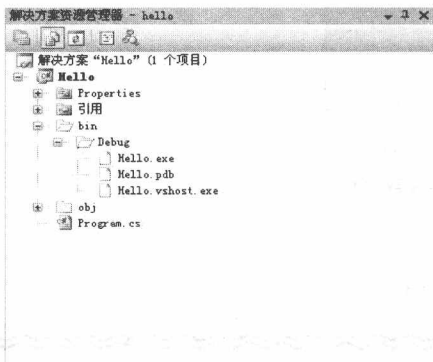


图 1-22 解决方案资源管理器



图 1-23 属性设置栏

● 窗体

C#的空窗体就像一个容器，是进行应用程序设计的场所。应用程序的界面设计就是在其上添加各种控件以及通过对它进行各种外观修饰来完成的。窗体由标题栏和下方空白设计区组成，如图 1-24 所示。由图可见，窗体标题栏右方依次排列最小化、最大化和关闭按钮。

● 代码窗口

代码窗口，顾名思义就是程序代码输入的场所。因为在 C#中，设计用户界面直接在窗体中进行，运行结果和设计样板完全一致。当组件被放到窗体上时，C#会自动生成大部分的用户界面代码，代码编辑只是在它生成的框架中加入完成所需功能的程序段而已。双击 Form1 的状态行使代码窗口可见。如图 1-25 所示显示了空窗体 Form1 的代码窗口。

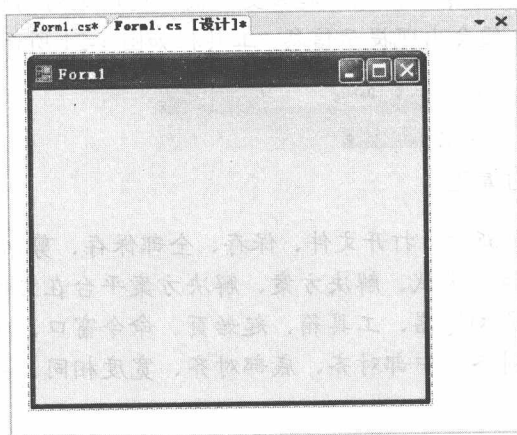


图 1-24 窗体

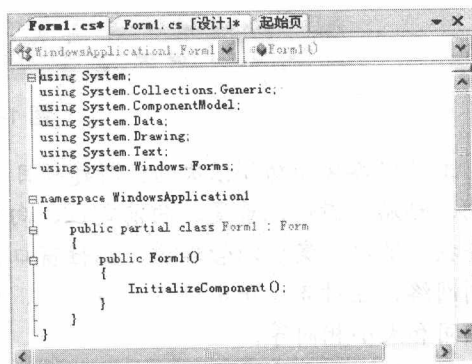


图 1-25 代码编辑窗口

● 工具箱

工具箱是 C#中最重要的组件之一，它位于整个屏幕的左边，如图 1-26 所示。在工具箱中包含了“所有 Windows 窗体”、“所有控件”、“容器”、“菜单和工具栏”、“数据”、“组件”、“打印”、“对话框”、Crystal Reports 和“常规”控件。

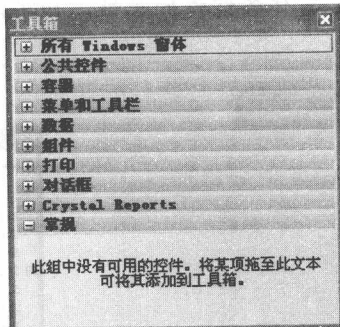


图 1-26 工具箱

四、知识拓展

(1) 新建控制台应用程序，完成猜数字游戏。关键代码如下：

```
using System;
public class guessNumber
{
    public static void Main()
    {
        int i_random=new Random().Next(100);
        int i_guess=0;
        int i_count=0;
        Console.WriteLine("猜一猜这样一个随机数,它的范围是在零到一百之间.");
        for(i_count=0;i_count<10;i_count++)
        {
            i_guess=int.Parse(Console.ReadLine());
            if(i_guess<i_random)
            {
                Console.WriteLine("太小了,大一点儿好些.");
            }
            if(i_guess>i_random)
            {
                Console.WriteLine("太大了,小一点儿好些.");
            }
        }
        if(i_guess==i_random&& i_count!=10)
        {
            Console.WriteLine("不错,不错,猜对了!");
        }
        else if(i_guess==i_random&& i_count==10)
        {
            Console.WriteLine("在这功败垂成的时候,你猜对了,更加幸运!");
        }
        else if(i_count==10&& i_guess!=i_random)
        {
            Console.WriteLine("对不起,你已经猜了十次,不能再猜了.");
        }
        Console.WriteLine(i_random);
    }
}
```

(2) 新建控制台应用程序，完成身份验证的功能。关键代码如下：

```
using System;
class validate
{
    public static void Main()
    {
        string str_wrench="ilovemyfamily";
        string str_next=" ";
    }
}
```