

高职高专“十一五”计算机类课程改革规划教材

COMPUTER

C#面向对象程序设计

OBJECT-ORIENTED PROGRAMMING WITH C#

王晶晶 主编



赠电子课件



机械工业出版社
CHINA MACHINE PRESS

高职高专“十一五”计算机类课程改革规划教材

C#面向对象程序设计

主 编 王晶晶
副主编 肖秋霞
参 编 张 坤 吴 岩
主 审 章五一



机械工业出版社

本书结合了高职高专教育的特点,系统讲解了C#语言及其程序设计过程。全书共分为11章,从基本概念和实际应用出发,由浅入深、循序渐进地讲述了C#语言基础语法、结构化程序设计、面向对象程序设计、Windows应用程序开发和ADO.NET访问数据库等内容,每一部分结合典型实例,让学生在学的过程中动手操作,在学好理论知识的同时强化专业技能训练,最后又通过学生信息管理系统这一综合实例,讲述了使用C#开发信息管理系统的过程和技术。

本书既可作为高职高专院校计算机及相关专业的教材,也可作为广大工程技术人员及计算机爱好者的参考用书。

为方便教学,本书配备电子课件等教学资源。凡选用本书作为教材的教师均可登录机械工业出版社教材服务网 www.cmpedu.com 免费下载。如有问题请致信 cmpgaozhi@sina.com,或致电010-88379375联系营销人员。

图书在版编目(CIP)数据

C#面向对象程序设计/王晶晶主编. —北京:机械工业出版社,2010.4
高职高专“十一五”计算机类课程改革规划教材
ISBN 978-7-111-30056-4

I. ①C… II. ①王… III. ①C语言—程序设计—高等学校:技术学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第041859号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)
策划编辑:王玉鑫 责任编辑:李大国 版式设计:张世琴
封面设计:王伟光 责任校对:陈延翔 责任印制:乔宇
北京汇林印务有限公司印刷
2010年5月第1版第1次印刷
184mm×260mm·15.75印张·390千字
0001-4000册
标准书号:ISBN 978-7-111-30056-4
定价:28.00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

网络服务

社服务中心:(010)88361066

门户网:<http://www.cmpbook.com>

销售一部:(010)68326294

教材网:<http://www.cmpedu.com>

销售二部:(010)88379649

读者服务部:(010)68993821

封面无防伪标均为盗版

前 言

随着 .NET Framework 2.0 的发布, 微软公司推出的新一代软件开发工具 Visual Studio 2005 备受 IT 行业的关注, 从而迎来了应用程序开发的新时代。于是, C# 及相关的 .NET Framework 开发环境成为备受推崇的新技术之一, 越来越多的软件开发人员开始学习该技术, 高等学校也将 C# 作为计算机专业重要的语言课程。

本书结构清晰, 内容详实, 案例丰富, 讲解透彻, 可作为高职高专计算机专业程序设计的教材。本书以实际工作过程中所需要的知识和技能为出发点, 注重知识的过渡与衔接, 精心设计实例, 强化学生的动手能力, 提高学生的专业技能。

本书共分为 11 章, 从基本概念和实际应用出发, 由浅入深、循序渐进地讲述了 C# 语言基础语法、结构化程序设计、面向对象程序设计、Windows 应用程序开发和 ADO.NET 访问数据库等内容, 每一部分结合典型实例, 让学生在学的过程中动手操作, 在学好理论知识的同时强化专业技能训练, 最后又通过学生信息管理系统这一综合实例, 讲述了使用 C# 开发信息管理系统的过程和技术。学习完本书, 学生能够掌握 C# 的基本知识和技术, 并能编写 Windows 桌面应用程序。本书的结构安排如下:

第 1 章 C# 概述, 介绍了 C# 的发展和特点, Visual Studio 2005 集成开发环境及运用该工具开发简单的控制台应用程序和 Windows 应用程序。

第 2 章 数据类型、运算符与表达式, 介绍了 C# 中数据类型、类型转换的规则和语法及运算符与表达式。

第 3 章 结构化程序设计, 介绍了结构化程序设计中 3 种基本结构: 顺序结构、选择结构和循环结构。

第 4 章 面向对象编程基础, 介绍了面向对象编程的基本概念, 包括: 类和对象的定义、构造函数与析构函数、this 关键字、属性、方法、静态和实例成员。

第 5 章 继承、多态与接口, 介绍了继承、多态性、抽象类和抽象方法、接口。

第 6 章 委托、事件及异常处理, 介绍了委托、事件和异常处理。

第 7 章 Windows 程序开发基础, 介绍了 Windows 应用程序中所用到的控件及其属性和方法、事件处理等。

第 8 章 Windows Forms 高级应用, 介绍了菜单栏、工具栏、状态栏和 MDI 应用程序。

第 9 章 文件与通用对话框, 介绍了文件与目录管理、文件的读和写及通用对话框。

第 10 章 ADO.NET 访问数据库, 介绍了 ADO.NET 技术、访问数据库的步骤及常用的 ADO.NET 对象的属性和方法。

第 11 章 学生信息管理系统开发, 介绍了系统开发的整个过程, 包括系统分析、详细设计、系统实现和部署应用程序。

本书由王晶晶主编, 肖秋霞任副主编。编写分工如下: 第 1~3 章由肖秋霞编写, 第 4、9 章由张坤编写, 第 5、6 章由吴岩编写, 第 7、8、10、11 章由王晶晶编写。全书由王晶晶统稿, 章五一教授在百忙之中对全书进行了认真审阅, 并提出了许多宝贵的意见和建议, 在

此表示衷心感谢!

由于编者水平有限,编写时间仓促,书中难免有错漏之处,敬请广大读者批评指正,以便下次修订时完善。

编 者

目 录

前言	
第1章 C#概述	1
1.1 C#的发展	1
1.2 C#的特点	1
1.3 C#能编写的应用程序	2
1.4 开发工具	2
1.5 控制台应用程序	18
1.6 可视化程序开发初探	22
习题1	24
第2章 数据类型、运算符与表达式	26
2.1 数据类型	26
2.2 类型转换	31
2.3 运算符和表达式	35
习题2	40
第3章 结构化程序设计	42
3.1 顺序结构	42
3.2 选择结构	42
3.3 循环结构	45
3.4 数组	51
习题3	57
第4章 面向对象编程基础	59
4.1 类与对象	59
4.2 构造函数与析构函数	62
4.3 this 关键字	65
4.4 属性	68
4.5 方法	71
4.6 静态和实例类成员	79
习题4	80
第5章 继承、多态与接口	82
5.1 继承	82
5.2 多态性	87
5.3 抽象类和抽象方法	91
5.4 接口	95
习题5	100
第6章 委托、事件及异常处理	102
6.1 委托	102
6.2 事件	108
6.3 异常处理	111
习题6	119
第7章 Windows 程序开发基础	121
7.1 窗体	121
7.2 控件概述	124
7.3 Label 控件	126
7.4 TextBox 控件	126
7.5 Button 控件	131
7.6 GroupBox 控件	133
7.7 RadioButton 控件	134
7.8 CheckBox 控件	136
7.9 ListBox 控件	138
7.10 ComboBox 控件	142
7.11 Timer 控件	145
7.12 LinkLabel 控件	147
7.13 PictureBox 控件	148
习题7	148
第8章 Windows Forms 高级应用	150
8.1 菜单	150
8.2 工具栏	154
8.3 状态栏	157
8.4 MDI 应用程序	159
习题8	164

第9章 文件与通用对话框	165	10.10 CommandBuilder 对象	208
9.1 文件与目录管理	165	10.11 使用存储过程	209
9.2 文件的读和写	170	10.12 Windows 应用程序访问	
9.3 通用对话框	175	数据库	214
习题9	185	习题10	228
第10章 ADO.NET 访问数据库	186	第11章 学生信息管理系统开发	229
10.1 ADO.NET 简介	186	11.1 系统分析	229
10.2 访问数据库的步骤	187	11.2 详细设计	232
10.3 连接数据源	187	11.3 系统实现	236
10.4 操作数据库中的数据	190	11.4 部署应用程序	242
10.5 带参数的 Command 对象	193	习题11	243
10.6 DataReader 对象	195	附录	244
10.7 DataSet 对象	197	附录A 参考答案	244
10.8 DataAdapter 对象	201	附录B C#关键字	245
10.9 DataReader 与 DataSet 的		参考文献	246
区别	207		

第 1 章 C# 概述

本章介绍 C# 的特点、开发环境和创建控制台应用程序的方法，在初步了解 C# 之后，学习安装 C# 的集成开发环境，并进行简单的控制台程序的编写。

1.1 C# 的发展

1995 年，Sun 公司推出了面向对象的程序开发语言 Java，该语言所具备的跨平台、跨语言的特点使 Java 逐渐成为企业级应用系统开发的首选工具。不久，微软公司推出了基于 Java 语言的编译器 Visual J++，Visual J++ 很快从 1.1 版本升级到了 6.0 版本，Visual J++ 6.0 集成在 Visual Studio 6.0 中。Visual J++ 虽然具有强大的开发功能，但主要应用在 Windows 平台的系统开发中，Sun 公司认为这违反了 Java 跨平台开发的特点，对微软提出了诉讼，这使微软公司处于极其被动的局面，为了改变这种局面，微软公司决定推出 .NET 计划和开发语言 C#。

微软公司在 2000 年 6 月举行的“职业开发人员技术大会”上正式发布了 C# 语言。其英文名为“C-Sharp”。微软公司对 C# 的定义是：“C# 是一种类型安全的、现代的、简单的，由 C 和 C++ 衍生出来的面向对象的编程语言，它是牢牢根植于 C 和 C++ 语言之上的，并可立即被 C 和 C++ 开发人员所熟悉。C# 的目的就是综合 Visual Basic 的高生产率和 C++ 的行动力。”

1.2 C# 的特点

C# 是一种简单、灵活且功能强大的程序设计语言，使程序员能够快速开发多种应用程序。由于它彻底采用了面向对象程序设计思想，因此无论是最普通的应用程序还是大型的商业软件，C# 都是最合适的选择。

简而言之，C# 有以下四个特点：

1. 简单

C# 继承了 C 和 C++ 的优点，并对其缺点进行了改善，使得语言更加简单。C# 同时摒弃了其他编程语言（例如 C++ 和 Java）中的一些复杂性和缺陷，使得即使没有编程基础的程序员也能有效地减少开发过程中的错误。

2. 面向对象

C# 具有面向对象程序设计语言所应有的一切特性：封装、继承和多态。通过面向对象的强大功能，C# 使得程序员的编程效率得到极大的提高，缩短了应用程序的开发周期。

3. 与 Web 紧密结合

C# 对网络中结构化数据传送的标准——XML 提供了很好的支持，程序员能够利用简单的 C# 语言结构方便地开发 XML Web Service，有效地处理网络中的各种数据。

4. 基于 .NET Framework

.NET Framework 为用 C#编写的应用程序提供了安全性保障和错误处理机制。

1.3 C#能编写的应用程序

C#能编写的应用程序主要有：

1. Windows 应用程序

Windows 应用程序有用户很熟悉的 Windows 外观和操作方式，使用 .NET Framework 的 Windows Forms 模块就可以生成这种应用程序。Windows Forms 模块是一个控件（例如按钮、工具栏、菜单等）库，其中的控件可以用于建立 Windows 用户界面（UI）。

2. Web 应用程序

Web 应用程序是由 Web 页组成的，可以通过任何 Web 浏览器查看。.NET Framework 包含一个动态生成 Web 内容的强大系统，这个系统叫做 Active Server Pages. NET（ASP.NET），可以使用 C#通过 Web Forms 创建 ASP.NET 应用程序。

3. Web 服务

Web 服务是创建各种分布式应用程序的新方式，使用 Web 服务可以通过 Internet 虚拟交换数据。无论使用什么语言创建 Web 服务，也无论 Web 服务驻留在什么系统上，都使用一样简单的语法。

1.4 开发工具

用户可以使用 Windows 自带的“记事本”来进行程序的开发。但是，其工作效率是很低的。为了解决该问题，集成开发环境（Integrated Development Environment，IDE）应运而生。本书采用的是 Visual Studio 2005 集成开发环境。

下面首先介绍如何安装 Visual Studio 2005。

1.4.1 安装 Visual Studio 2005

1. 准备工作

（1）硬件

处理器：主频在 600MHz 以上，建议采用主频在 1GHz 以上的 Pentium 处理器。

系统内存：最低要求为 192MB，推荐为大于 256MB。

硬盘：若不安装 MSDN，安装驱动器上要有 2GB 可用空间，系统驱动器上要有 1GB 可用空间。若安装 MSDN，在完全安装 MSDN 的安装驱动器上要有 3.8GB 的可用空间；在进行默认 MSDN 安装的安装驱动器上要有 2.8GB 的可用空间，在系统驱动器上要有 1GB 可用空间。

显示器：最低要求为 800 × 600 像素、256 色，推荐使用 1024 × 768 像素、增强 16 位颜色。

（2）运行环境（操作系统）

对于 Microsoft Windows 2000 系列操作系统，需要安装 SP4。

对于 Microsoft Windows XP 系列操作系统，需要安装 SP2。

对于 Microsoft Windows Server 2003，需要安装 SP1。

浏览器要求 IE5.5 或更新版本。

(3) Visual Studio 2005 安装程序。Microsoft Visual Studio 2005 Standard Edition 或 Microsoft Visual Studio 2005 Professional Edition。如未特别说明，本书中出现的截屏图均为 Microsoft Visual Studio 2005 Professional Edition。

2. 安装 Visual Studio 2005

(1) 启动安装程序。从光盘或安装程序目录启动 Setup.exe，如图 1-1 所示。



图 1-1 启动安装程序

(2) 安装 Visual Studio 2005。单击“安装 Visual Studio 2005”，进入 Visual Studio 2005 的安装向导，如图 1-2 所示。安装程序开始加载安装组件。

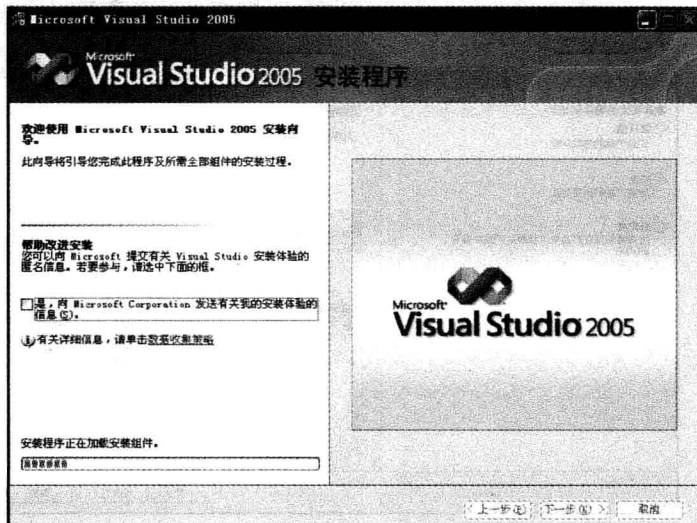


图 1-2 安装向导

在出现如图 1-2 所示的画面前，会弹出一个小窗口——安装程序在解压安装文件。如果计算机硬件配置较低，可能需要的时间会长些，请耐心等待。

(3) 安装程序加载安装组件。安装程序加载安装组件完成后，“下一步”按钮由灰色变为可以单击状态。单击“下一步”按钮进入安装程序起始页，如图 1-3 所示。

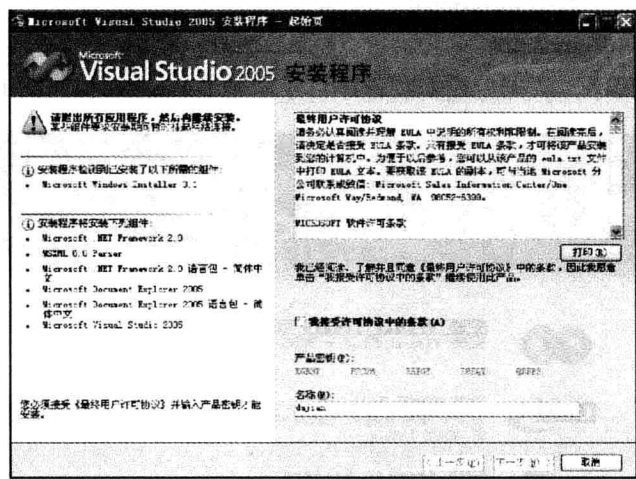


图 1-3 安装程序起始页

在起始页的右侧是“最终用户许可协议”，选中“我接受许可协议中的条款”复选框，并输入产品密钥，然后单击“下一步”按钮，进入安装选项页。

(4) 安装选项。安装选项用于选择要安装的功能，可以选择“默认值”、“完全”和“自定义”三种安装方式，并可以指定 Visual Studio 2005 的安装路径，如图 1-4 所示。建议选择“默认值”选项。如果用户对 Visual Studio 2005 所包含的各种功能都比较了解，则可以根据需要选择“自定义”选项，只安装需要的功能，以节省硬盘空间。

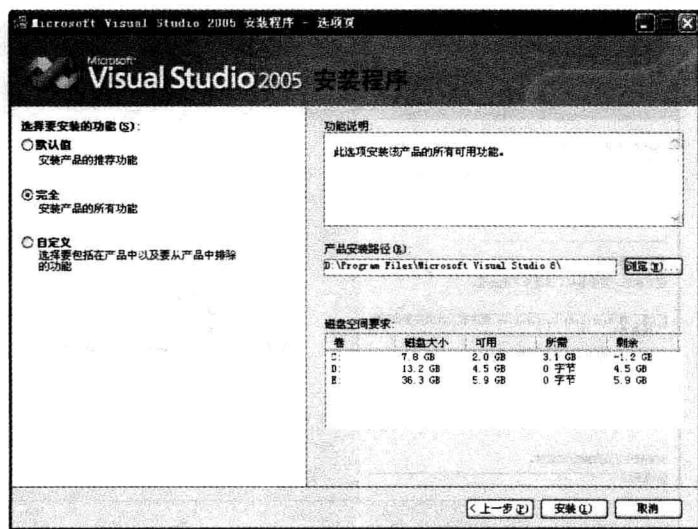


图 1-4 安装选项

可以根据各磁盘的使用情况,通过单击“浏览”按钮来选择产品的安装路径。

(5) 开始在计算机上安装 Visual Studio 2005 的组件。选择好安装路径后,单击“安装”按钮,安装程序将开始安装 Visual Studio 2005 的组件,如图 1-5 所示。根据目标计算机上已经安装的 Visual Studio 2005 的组件的多少,在这里要进行安装的项目会有所不同。如果安装 Visual Studio 2005 的计算机中没有安装 .NET Framework 2.0、Microsoft XML 等运行 Visual Studio 2005 所必需的组件,则安装程序会将这些组件自动安装到目标计算机上。

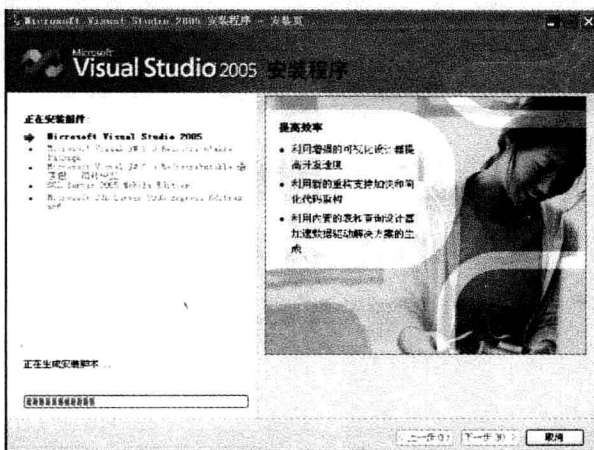


图 1-5 安装 Visual Studio 2005 的组件

根据选择安装的组件和计算机中已经安装的运行 Visual Studio 2005 所必需组件的多少,安装过程所需要的时间长短会有所不同。安装过程中,右侧窗格中的动画显示了 Visual Studio 2005 的新特性,可以边等待边浏览这些新特性。

(6) 安装完毕。系统安装完毕后会提示安装成功,如图 1-6 所示。单击“完成”按钮返回到最初的安装程序界面。

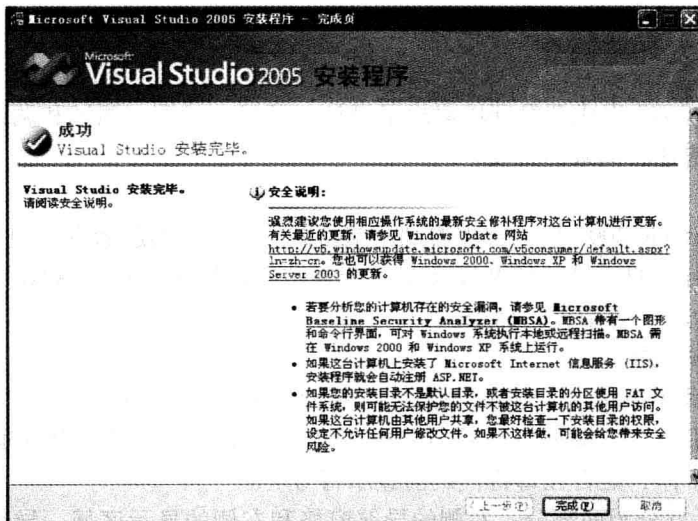


图 1-6 系统安装完毕

(7) 安装 Visual Studio 帮助——MSDN。MSDN 是 Microsoft Software Developer Network 的简称。Visual Studio 2005 的 MSDN 中包含 Visual Studio 帮助信息和各个开发语言包的相关知识库。可以说 MSDN 是 Visual Studio 开发最全面、也是最权威的技术文档。

在安装完成 Visual Studio 2005 后，再次返回安装程序界面后，如图 1-7 所示。单击“安装产品文档”，开始安装 MSDN。其安装过程与前面类似，这里不再赘述。



图 1-7 安装 MSDN

由于 MSDN 包含了整个 Visual Studio 2005 所有功能的帮助文档，所以比较庞大，建议只选择安装需要的部分。在以后需要时可以重新安装初次安装时未选中的部分。如果硬盘容量足够大，也可以选择完全安装，便于学习。

(8) 安装结束。安装完成后，单击“退出”按钮，结束安装过程。至此，Visual Studio 2005 安装成功。

1.4.2 集成开发环境简介

Visual Studio 2005 集成开发环境集成了设计、开发、编辑、测试和调试等多种功能，方便开发人员进行快速、高效的开发。

1. 起始页

打开 Visual Studio 2005 后，首先看到的的就是图 1-8 所示的起始页。使用起始页可以轻松访问或创建项目、阅读最新的开发文章。若要访问起始页，在“视图”菜单中选择“其他窗口”，然后单击“起始页”命令。

2. “选项”对话框

在“工具”菜单下单击“选项”命令，便可访问到如图 1-9 所示的“选项”对话框。

在该对话框中，可以根据自己的需要配置开发环境，比如可以设置项目的默认保存位置、设置代码的字体、是否在每行代码前显示行号等。

“选项”对话框分为两部分：左侧的导航窗格和右侧的显示区域。导航窗格中的树控件包括如下文件夹节点：“环境”、“项目和解决方案”、“源代码管理”以及“文本编辑器”

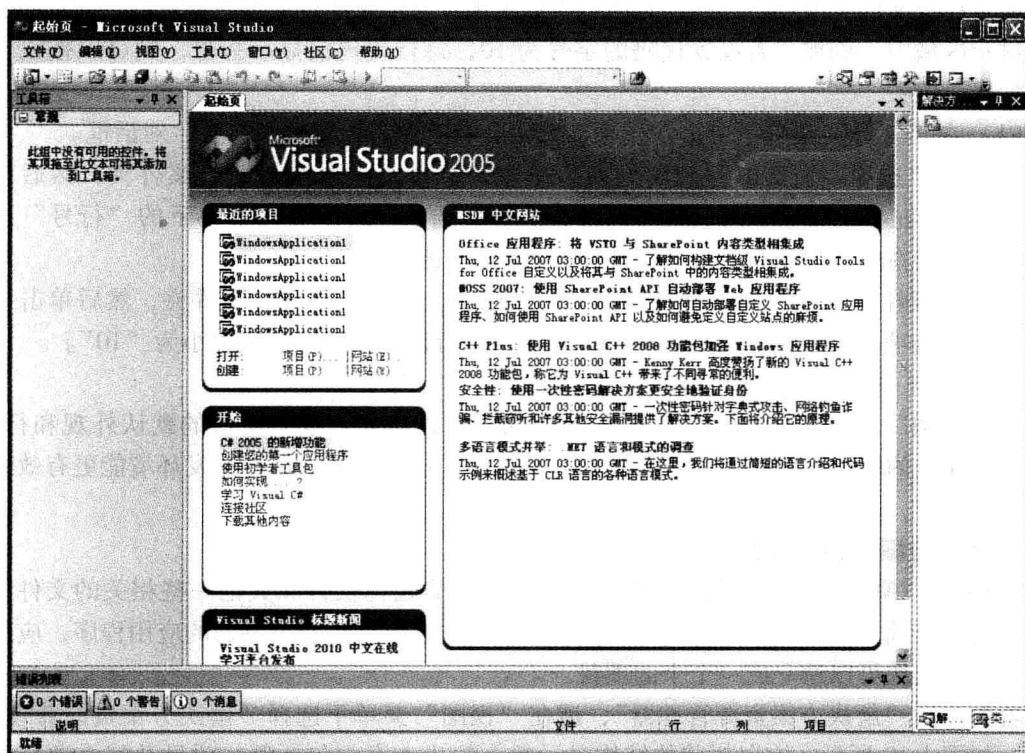


图 1-8 Visual Studio 2005 起始页

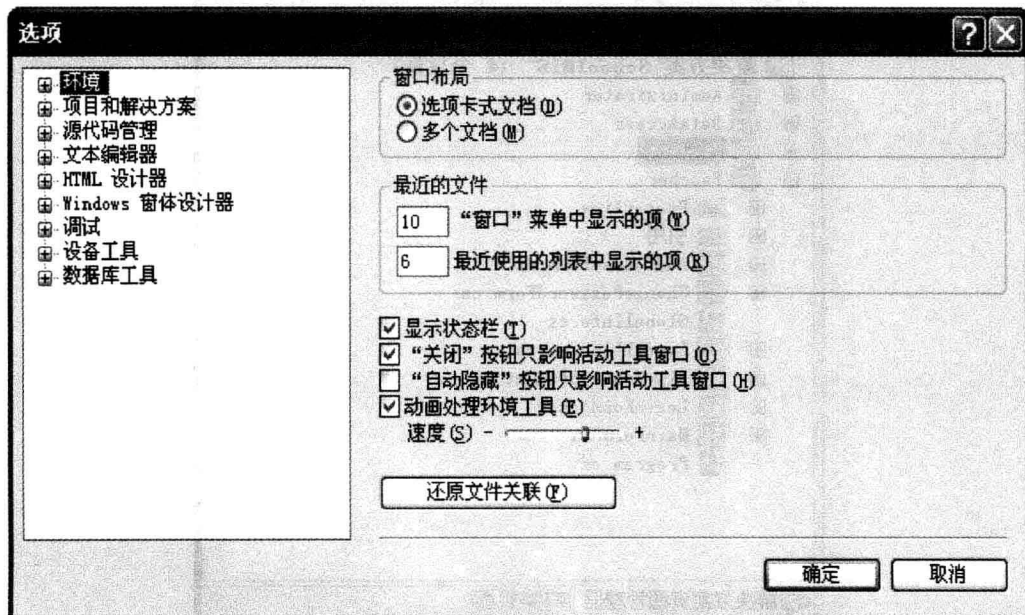


图 1-9 “选项”对话框

等。展开任意文件夹节点可以列出它所包含的选项页。当选择特定页的节点时，其选项会出现在显示区域中。例如，若设置代码的字号为 10，每行代码前显示行号，则可以按以下步骤来加以设置：

(1) 在“工具”菜单下单击“选项”命令，打开图 1-9 所示“选项”对话框。

(2) 在“选项”对话框左侧的导航窗格中，单击“文本编辑器”文件夹，然后单击“C#”选项，在右侧的显示区域便可列出其所有的选项。选中“显示”下的“行号”这一复选框，这样在编写代码时每行代码前便会自动添加行号。

(3) 继续在“选项”对话框左侧的导航窗格中，单击“环境”文件夹，然后单击“字体和颜色”选项，在右侧的显示区域中可对字体进行设置，这里设置大小为“10”。

(4) 单击“确定”按钮，保存所做的设置。

在“选项”对话框中，还可以进行其他的设置，比如：改变窗口的默认外观和行为，创建常用命令的快捷方式等，读者可以参照上述例子进行设置，使得开发环境能更有效地为你服务。

3. 解决方案资源管理器

在 C# 中，项目是一个独立的编程单位，可以包含窗体文件和其他一些相关的文件。若多个项目就组成了一个解决方案。也就是说，解决方案就是用户要创建的应用程序，应用程序下的各个模块可以建立成一个个的项目。

若要访问“解决方案资源管理器”，请在“视图”菜单下选择“解决方案资源管理器”命令，如图 1-10 所示。它以树状的结构显示整个解决方案中包括哪些项目及每个项目的组成信息，在该对话框中可以对项目进行修改和执行其他的管理任务。



图 1-10 “解决方案资源管理器”对话框

在实际开发应用程序时，常常需要单个项目的运行和调试。此时需要指定某个项目为“启动项目”。默认情况下，将在解决方案中创建的第一个项目指定为启动项目，但也可以在生成或运行解决方案时设置需要先运行的项目。设置单个项目为启动项目的方法：在项目上单击鼠标右键，选择“设为启动项目”即可，如图 1-11 所示。启动 Visual Studio 调试器时，会自动运行启动项目。

如果有多个启动项目，则树视图中的解决方案节点会以粗体字显示。设置多个项目为启动项目的方法：在解决方案上单击鼠标右键，选择“设置启动项目...”命令，如图 1-12 所示，然后在弹出的“解决方案属性页”窗口左侧的树视图中选择“启动项目”命令，在右侧选中“多启动项目”单选按钮，然后设置项目的操作属性为“启动”即可，如图 1-13 所示。

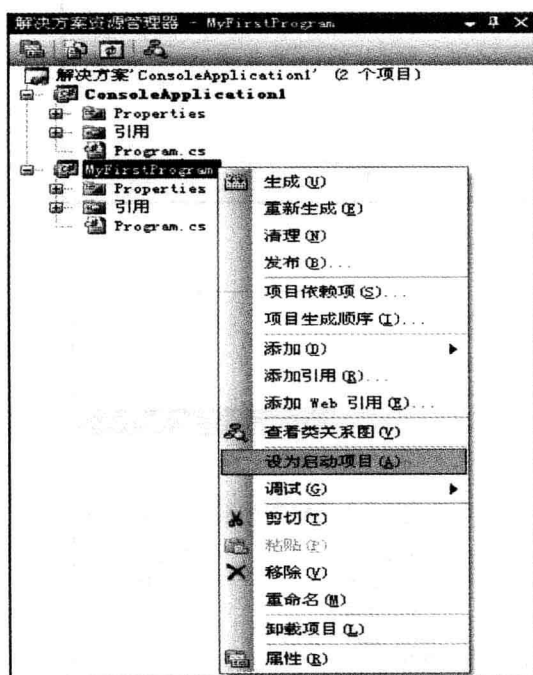


图 1-11 将单个项目设为启动项目

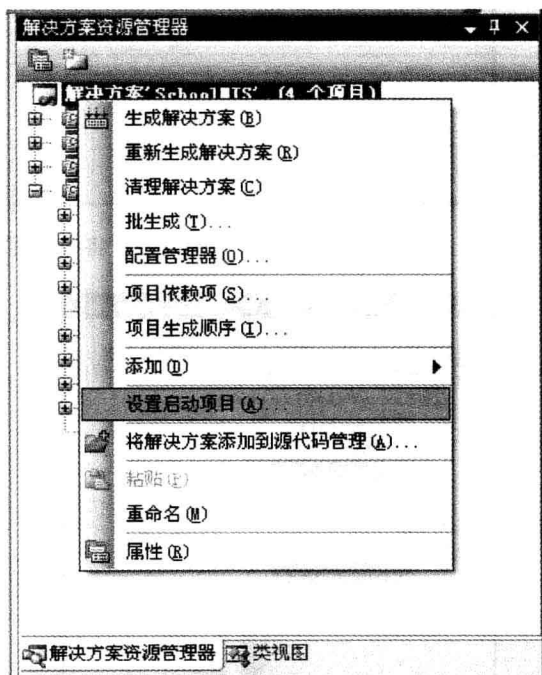


图 1-12 将多个项目设为启动项目 (1)

可以在单个项目中选择多个项或跨项目选择多个项。当希望执行批操作（例如同时打开多个文件进行编辑），或者希望确定或编辑两个以上解决方案项的共同可用属性时，请选择多个项。当选择多个项时，可用的命令表示对所有选定项通用的命令。这样就可以方便地一次性设置多个项目文件的共同属性了，如图 1-14 和图 1-15 所示。

在“解决方案资源管理器”的最上面有一些小图标，它们就是“解决方案资源管理器”的工具栏图标，见表 1-1。它们代表了一些常用的命令。由于选定项目的不同决定了工具栏图标的不同，因而此列表只表示在使用“解决方案资源管理器”时可能遇到的一部分图标。

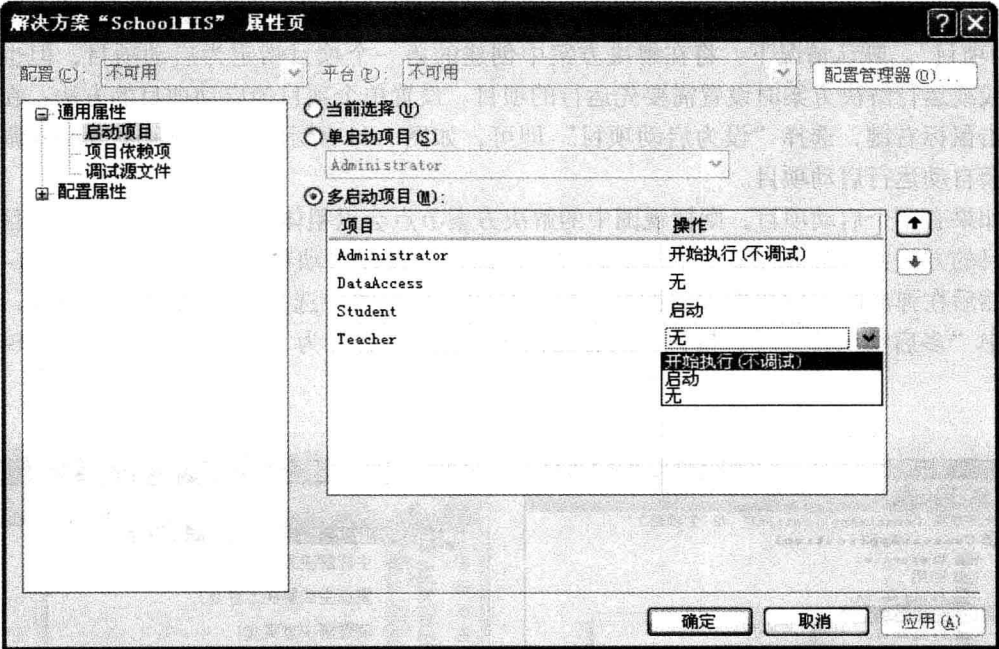


图 1-13 将多个项目设为启动项目（2）

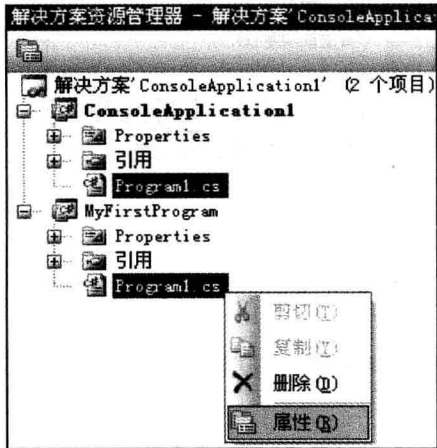


图 1-14 设置多个项目文件的共同属性（1）

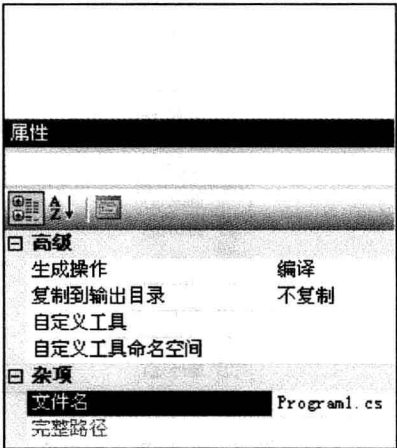


图 1-15 设置多个项目文件的共同属性（2）

表 1-1 “解决方案资源管理器” 工具栏图标

图 标	含 义	详 细 解 释
	属性	显示树视图中所选项的相应属性对话框
	显示所有文件	显示所有项目项，包括那些已经被删除的和正常情况下隐藏的项
	刷新	刷新所选项或解决方案中的项的状态