

中央财政“支持高等职业学校提升专业服务产业发展能力”项目建设成果

C# 程序设计

——理实一体化教学课程

C# CHENGXU SHEJI

LISHI YITIHUA JIAOXUE KECHENG

唐 权 梁 琰◎主编



西南交通大学出版社

[Http://press.swjtu.edu.cn](http://press.swjtu.edu.cn)

中央财政“支持高等职业学校提升专业服务产业发展能力”项目建设成果

C#程序设计——理实一体化教学课程

主 编 唐 权 梁 琰

副主编 韩文智 骆文亮

参 编 许 俊 朱 倩 陈 印

马红春 陈 倬

西南交通大学出版社

· 成 都 ·

图书在版编目 (C I P) 数据

C#程序设计:理实一体化教学课程 / 唐权, 梁琰
主编. —成都:西南交通大学出版社, 2014.3
ISBN 978-7-5643-2913-6

I. ①C… II. ①唐… ②梁… III. ①C语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 027323 号

C#程序设计——理实一体化教学课程

主编 唐权 梁琰

责任编辑	李芳芳
助理编辑	宋彦博
封面设计	墨创文化
出版发行	西南交通大学出版社 (四川省成都市金牛区交大路 146 号)
发行部电话	028-87600564 028-87600533
邮政编码	610031
网 址	http://press.swjtu.edu.cn
印 刷	四川五洲彩印有限责任公司
成品尺寸	185 mm × 260 mm
印 张	10
字 数	250 千字
版 次	2014 年 3 月第 1 版
印 次	2014 年 3 月第 1 次
书 号	ISBN 978-7-5643-2913-6
定 价	24.00 元

图书如有印装质量问题 本社负责退换
版权所有 盗版必究 举报电话: 028-87600562

前 言

C#是目前广泛使用的一种编程语言,可用于开发 Windows 应用程序、Web 应用程序和移动应用程序。在高校中,“C#程序设计”是计算机专业的一门专业基础课,其目标主要是培养学生分析问题、解决问题和实际动手操作的能力。本书打破传统的理论教学模式,采用理论和实践相结合的方式,从 C# 语言的基础讲起,重点讲解了结构化程序设计、面向对象基础、可视化编程、数据库处理技术和文件流等内容,并对每一部分的细节进行了详细的说明。

通过对本书的学习,学生可以掌握面向对象程序设计和可视化程序设计的基本方法。书中提供了大量的 C# 应用程序实例,并对每个实例的操作步骤进行了详细的阐述。学生可以按照所述步骤,自己动手完成每一个实例,从而加强对实际动手能力的培养。

本书由四川职业技术学院唐权、梁琰担任主编,由韩文智、骆文亮担任副主编。具体的编写分工为:第 1 章由许俊编写,第 2 章由韩文智、朱倩编写,第 3 章由骆文亮编写,第 4 章由陈倬编写,第 5 章由梁琰编写,第 6 章由陈印、马红春编写,第 7 章由唐权编写。

在本书的编写过程中,我们参考了大量文献资料,在此向相关文献作者表示衷心感谢!由于编者水平有限,加之编写时间比较仓促,书中难免有疏漏和不足之处,恳请同行专家及读者提出宝贵意见和建议。

编 者

2013 年 11 月

目 录

第 1 章 C#概述	1
1.1 当前流行的面向对象开发语言概述	1
1.2 C#的基础框架——.NET Framework	2
1.3 C#简介	4
1.4 C#开发工具 Visual Studio 简介	5
1.5 创建 C#控制台程序	13
1.6 创建 Windows 应用程序	19
第 2 章 C#语法知识	23
2.1 C#语言元素	23
2.2 变 量	24
2.3 数据类型	25
2.4 运算符与表达式	32
2.5 程序流程控制	38
2.6 异常处理	45
第 3 章 面向对象程序设计	48
3.1 面向对象程序设计的基本概念	48
3.2 对象与类	49
3.3 类的构造函数与析构函数	58
3.4 类构造函数的重载	63
3.5 继承性	66
3.6 多 态	68
第 4 章 数 组	73
4.1 一维数组	73
4.2 二维数组	75
4.3 多维数组	77
4.4 数组的属性与方法	77
第 5 章 Windows 应用程序	81
5.1 Windows 应用程序的开发环境	81
5.2 控件的属性和事件概述	84

5.3	窗体 (Form)	85
5.4	命令按钮控件 (Button)	85
5.5	标签控件 (Label)	87
5.6	文本框控件 (TextBox)	89
5.7	列表框控件 (ListBox)	92
5.8	组合框控件 (ComboBox)	95
5.9	单选按钮控件 (RadioButton)	98
5.10	复选框控件 (CheckBox)	100
5.11	面板控件 (Panel) 和分组框控件 (GroupBox)	104
5.12	选项卡控件 (TabControl)	107
5.13	菜单控件 (MenuStrip)	110
5.14	打开文件对话框控件 (OpenFileDialog)	112
5.15	定时器控件 (Timer)	114
5.16	综合应用	116
第 6 章	数据库处理技术 ADO.NET	122
6.1	ADO.NET 概述	122
6.2	数据库连接对象 (SqlConnection)	123
6.3	数据库命令执行对象 (SqlCommand)	125
6.4	数据阅读对象 (SqlDataReader)	129
6.5	数据集 (DataSet)	132
6.6	数据表格控件 (DataGridView)	135
第 7 章	文件和流	140
7.1	System.io 命名空间	140
7.2	stream 流读写文件	141
7.3	BinaryWriter 类写文件, BinaryReader 类读文件	144
7.4	StreamWriter 类写文件, StreamReader 类读文件	145
7.5	File 类与 FileInfo 类对文件的操作	147
7.6	Directory 类与 DirectoryInfo 类目录操作	149
参考文献		154

第 1 章 C#概述

【学习目标】

- ☞ 了解 .NET Framework 的工作原理；
- ☞ 认识 Visual C# 开发环境；
- ☞ 利用 C# 建立 Windows 应用。

【知识要点】

- 📖 .NET Framework 的工作原理；
- 📖 Visual Studio 2008 的安装；
- 📖 Visual Studio 2008 C# 的集成环境；
- 📖 控制台应用程序；
- 📖 Windows 窗体应用程序。

1.1 当前流行的面向对象开发语言概述

Java、C#、C++、PHP 是目前主流的面向对象编程语言。

1. Java 语言

Java 是面向对象、安全、跨平台的程序设计语言与环境，由 Sun 公司开发，是免费的、开源的，近些年来非常流行且稳定，未来生命周期较长。其语言风格较接近 C++ 与 C#，而最为人熟知的便是跨平台性。Java 的跨平台性已得到了广泛的认可，在计算机的各种平台、操作系统，以及手机、移动设备、智能卡、消费家电领域得到了广泛的应用。

2. C++ 语言

C++ 语言是当前应用最广泛的成熟、强大、复杂的程序设计语言。目前广泛使用的 Windows 或 Linux 操作系统的大部分内容均出自 C++ 程序员之手。C++ 非常强大，其代码经过编译后将成为计算机的二进制代码的可执行程序，所以在兼容性、性能上均十分优秀。

3. C# 语言

微软的 C# 语言就像是 C++、Java、Delphi 与 Visual Basic 的结合体，是新兴、易学、强大的程序设计语言。它更像 Java，完全面向对象，开发与运行都在 .NET Framework 环境中。

4. PHP 语言

PHP 语言是大多数门户网站、博客、论坛中采用的网页内部程序与数据处理的动态网页技术。PHP 是目前最流行、强大、稳健的动态网页开发脚本语言。

1.2 C#的基础框架——.NET Framework

1.2.1 .NET Framework 3.5 概述

.NET Framework 是一种技术,该技术支持生成和运行下一代应用程序和 XML Web Services。

1. .NET Framework 版本

.NET Framework 3.5 版是在 2.0 版和 3.0 版及其 Service Pack 的基础上构建的。.NET Framework 3.5 Service Pack 1 更新了 3.5 版程序集,并包含 2.0 版和 3.0 版的新 Service Pack。.NET Framework 的每个版本都可独立于更高版本进行安装,每个版本将自动安装早期版本(如果尚未安装这些早期版本)。此外,.NET Framework 3.5 SP1 还引入了客户端配置文件安装包,它只包含客户端应用程序所使用的程序集。应用程序无论面向的是 .NET Framework 2.0 版、3.0 版、3.5 版,还是客户端配置文件,该应用程序都将使用相同的程序集,并且与用户计算机上是否已更新这些程序集无关。.NET Framework 3.5 版为 2.0 版和 3.0 版中的技术引入了新功能,并以新程序集的形式引入了其他技术,如语言集成查询(LINQ),C#、Visual Basic 及 C++ 的新编译器,ASP.NET AJAX。.NET Framework 包括公共语言运行时和 .NET Framework 类库。

2. .NET Framework 的实现目标

- 提供一个一致的面向对象的编程环境,而无论对象代码是在本地存储和执行,还是在本地执行但在 Internet 上分布,或者是在远程执行。
- 提供一个将软件部署和版本控制冲突最小化的代码执行环境。
- 提供一个可提高代码执行安全性的代码执行环境。
- 提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。
- 使开发人员的经验在面对类型大不相同的应用程序(如基于 Windows 的应用程序和基于 Web 的应用程序)时保持一致。
- 按照工业标准生成所有通信,以确保基于 .NET Framework 的代码可与任何其他代码集成。

公共语言运行时和类库与应用程序之间以及与整个系统之间的关系如图 1.1 所示。

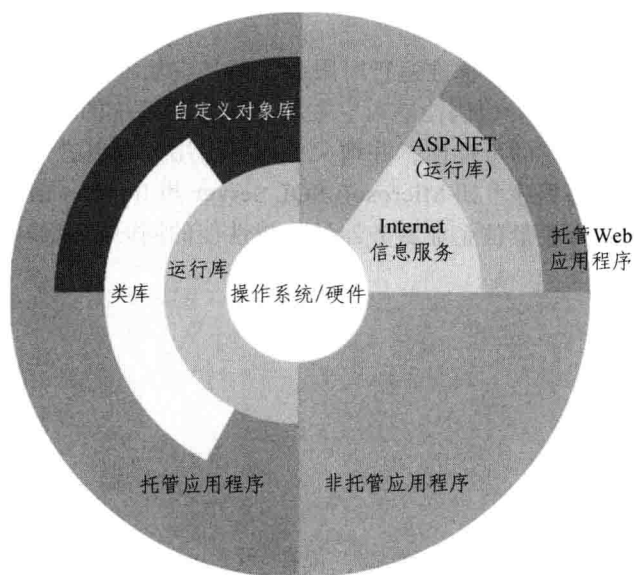


图 1.1 .NET Framework 框架图

1.2.2 .NET Framework 类库

.NET Framework 类库是一个与公共语言运行时紧密集成的可重用的类型集合。.NET Framework 类型使程序能够完成一系列常见编程任务，如字符串管理、数据收集、数据库连接以及文件访问等任务。除这些常规任务之外，类库还包括支持多种专用开发方案的类型。例如，可使用 .NET Framework 开发控制台应用程序、Windows GUI 应用程序(Windows 窗体)、ASP.NET 应用程序、Windows 服务等。

1.2.3 公共语言运行时

公共语言运行时管理内存、线程执行、代码执行、代码安全验证、编译以及其他系统服务。其作用如下：

(1) 运行时强制实施代码访问安全。

用户可以使用嵌在网页中的可执行文件，能够在屏幕上播放动画或唱歌，但不能访问他们的个人数据、文件系统或网络。这样，运行时的安全性功能就使通过 Internet 部署的合法软件能够具有特别丰富的功能，通过实现称为常规类型系统 (CTS) 的严格类型验证和代码验证基础结构来加强代码可靠性。

(2) 运行时的托管环境消除了许多常见的软件问题。

运行时自动处理对象布局并管理对对象的引用，在不再使用它们时将其释放。这种自动内存管理解决了两个最常见的应用程序错误：内存泄漏和无效内存引用。

(3) 运行时提高了开发人员的工作效率。

程序员可以用他们选择的开发语言编写应用程序，还能充分利用其他开发人员用其他语言编写的运行时、类库和组件。

(4) 运行时旨在增强性能。

尽管公共语言运行时提供许多标准运行时服务，但是它从不解释托管代码。一种称为实时 (JIT) 编译的功能使所有托管代码能够以它在其上执行的系统的本机语言运行。同时，内存管理器排除了出现零碎内存的可能性，并增大了内存引用区域以进一步提高性能。运行时可由高性能的服务器端应用程序 [如 Microsoft SQL Server 和 Internet 信息服务 (IIS)] 承载，在享受支持运行时承载的行业最佳企业服务器的优越性能的同时，能够使用托管代码编写业务逻辑。

1.3 C#简介

1.3.1 C#语言特点

C#是微软公司发布的一种新的编程语言，读作“C sharp”。它是一种安全的、稳定的、简单的、优雅的、由 C 和 C++衍生出来的、面向对象的编程语言，是专门为 .NET 的应用而开发的语言。C#继承了 C 语言的语法风格，同时又继承了 C++面向对象的特性。不同的是，C#的对象模型已经面向 Internet 进行了重新设计，使用的是 .NET 框架的类库；C#不再提供对指针类型的支持，使得程序不能随便访问内存地址空间，从而更加健壮；C#不再支持多重继承，避免了以往类层次结构中由多重继承带来的可怕后果。.NET 框架为 C#提供了一个强大的、易用的、逻辑结构一致的程序设计环境。并且，C#成为 ECMA 与 ISO 标准规范。

C#的特点有：

- (1) 语法简洁。C#不允许直接操作内存，去掉了指针操作。
- (2) 彻底的面向对象设计。C#具有面向对象语言所应有的一切特性——封装、继承和多态。
- (3) 与 Web 紧密结合。C#支持绝大多数 Web 标准，如 HTML、XML、SOAP 等。
- (4) 强大的安全机制。C#可以消除软件开发中的常见错误（如语法错误）。.NET 提供的垃圾回收器能够帮助开发者有效地管理内存资源。
- (5) 兼容性强。C#遵循 .NET 的公共语言规范 (CLS)，从而保证其能够与其他语言开发的组件兼容。
- (6) 灵活版本处理技术。C#本身内置了版本控制功能，从而使得开发人员可以更容易地开发和维护。
- (7) 完善的错误和异常处理机制。C#提供了完善的错误和异常处理机制，使程序在交付应用时能够更加健壮。

1.3.2 命名空间

命名空间在概念上与计算机文件系统中的文件夹有些类似。与文件夹一样，命名空间可使类具有唯一的完全限定名称，可以避免命名冲突。一个 C#程序包含一个或多个命名空间，每个命名空间可由程序员定义，也可作为之前编写的类库的一部分定义。

例如,命名空间 System 包括 Console 类,该类包含读取和写入控制台窗口的方法。System 命名空间也包含许多其他命名空间,如 System.IO 和 System.Collections。命名空间被用来最大限度地减少名称相似的类型和方法引起的混淆。

命名空间可以包含其他命名空间。这种划分方法的优点类似于文件夹。与文件夹不同的是,命名空间只是一种逻辑上的划分,而不是物理上的存储分类。

1.4 C#开发工具 Visual Studio 简介

1.4.1 Visual Studio 2008 开发环境

1. 安装 Visual Studio 2008 集成开发环境

1) 安装条件

安装 Visual Studio 2008 之前,需要了解安装其所必需的条件,并检查计算机的软硬件配置是否满足 Visual Studio 2008 开发环境的安装要求。具体要求如表 1.1 所示。

表 1.1 安装 Visual Studio 2008 所必需的条件

软硬件	描述
处理器	600 MHz 处理器,建议使用 1 GHz 处理器
RAM	512 MB,建议使用 1 GB 内存
可用硬盘空间	如果不安装 MSDN,系统驱动器上需要 1 GB 的可用空间,安装驱动器上需要 2 GB 的可用空间。如果安装 MSDN,则系统驱动器上需要 1 GB 的可用空间,完整安装 MSDN 的安装驱动器上需要 3.8 GB 的可用空间,默认安装 MSDN 的安装驱动器上需要 2.8 GB 的可用空间
CD-ROM 或 DVD-ROM 驱动器	必须使用
显示器	800×600 像素,256 色。建议使用 1024×768 像素,增强色 16 位
操作系统及所需补丁	Windows 2000 Service Pack 4、Windows XP Service Pack 2、Windows Server 2003 Service Pack 1 或更高版本

注意: Windows XP Home 不支持本地 Web 应用程序开发,只有 Windows 专业版和服务器版才支持本地 Web 应用程序开发。同时, Visual Studio 2008 还不支持 Windows 95、Windows 98、Windows Me 和 Microsoft Windows 2000 Datacenter Server 等平台。

2) 安装步骤

(1) 将 Visual Studio 2008 安装盘放到光驱中,光盘自动运行后会进入安装程序界面。如果光盘不能自动运行,可以双击 setup.exe 可执行文件,应用程序会自动跳转到如图 1.2 所示的“Visual Studio 2008 安装程序”界面。该界面上有 3 个安装选项,即安装 Visual Studio 2008、安装产品文档和检查 Service Release,一般情况下需安装前两项。



图 1.2 Visual Studio 2008 安装程序界面

(2) 单击“安装 Visual Studio 2008”，弹出如图 1.3 所示的安装向导界面。

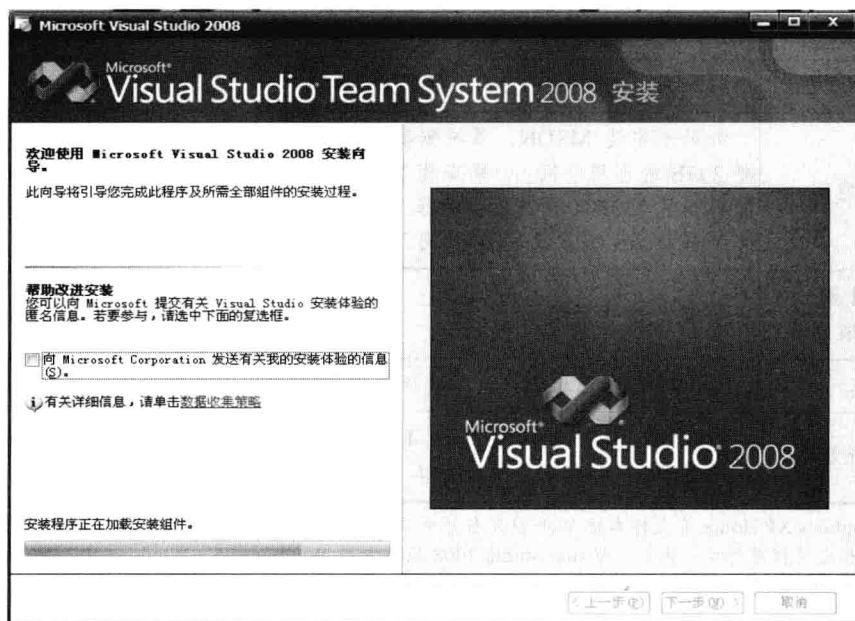


图 1.3 Visual Studio 2008 安装向导界面

(3) 单击“下一步”按钮，弹出如图 1.4 所示的“Microsoft Visual Studio 2008 安装程序-起始页”界面。界面左侧显示 Visual Studio 2008 安装程序的所需组件信息，右侧显示用户许可协议。



图 1.4 Visual Studio 2008 安装程序-起始页

(4) 选中“我已阅读并接受许可条款”单选按钮，单击“下一步”按钮，弹出如图 1.5 所示的“Microsoft Visual Studio 2008 安装程序-选项页”界面。用户可以选择要安装的功能和产品安装路径，一般使用默认设置即可（产品默认路径为“C:\Program Files\Microsoft Visual Studio 9.0”）。

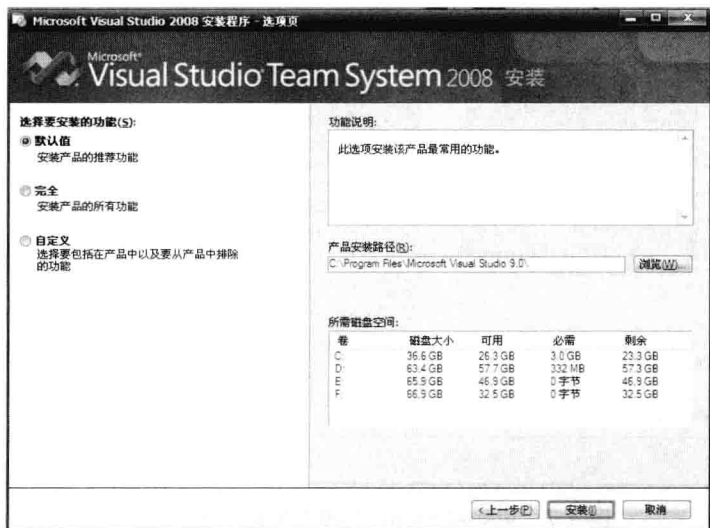


图 1.5 Visual Studio 2008 安装程序-选项页（一）

说明：在“选择要安装的功能”栏中，用户可以选择“默认值”“完全”和“自定义”3 种安装方式。如果选择“默认值”，安装程序将安装系统必备的功能；如果选择“完全”，安装程序将安装系统的所有功能；如果选择“自定义”，用户可以自由选择安装的项目，从而增强了安装程序的灵活性。本次安装选择“自定义”，语言工具选择“Visual C#”，如图 1.6 所示。

(5) 选择好产品安装路径之后，单击“安装”按钮，进入如图 1.7 所示的“Microsoft Visual Studio 2008 安装程序-安装页”界面，显示正在安装组件。

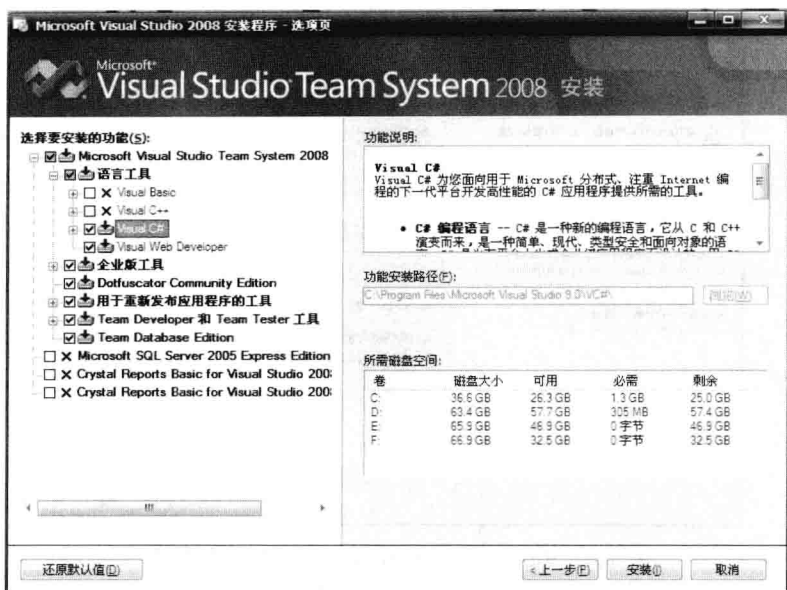


图 1.6 Visual Studio 2008 安装程序-选项页 (二)

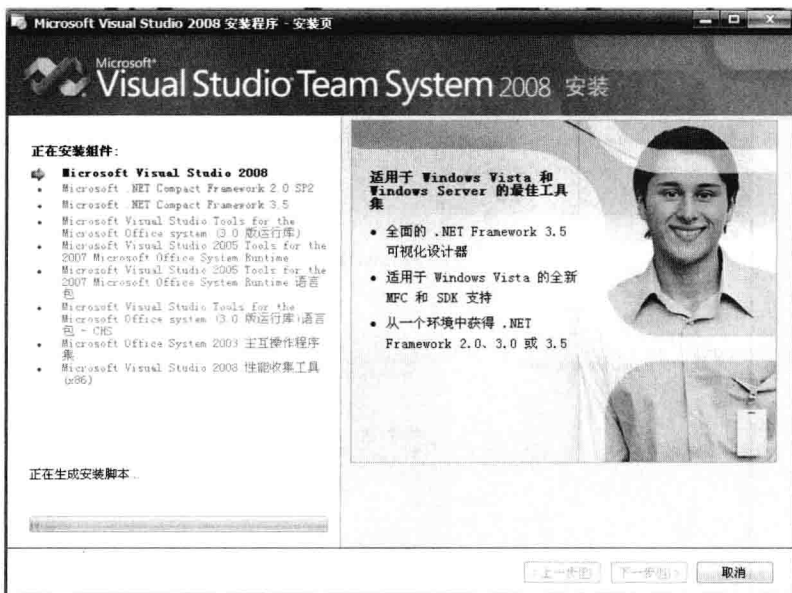


图 1.7 Visual Studio 2008 安装程序-安装页

(6) 安装完毕后, 单击“下一步”按钮, 弹出“Microsoft Visual Studio 2008 安装程序-完成页”界面, 单击“完成”按钮, 完成 Visual Studio 2008 集成开发环境的安装。

2. 启动 Visual Studio 2008 集成开发环境

(1) 选择“开始”/“程序”/Microsoft Visual Studio 2008/Microsoft Visual Studio 2008 命令, 如果用户是第一次使用 Visual Studio 2008 开发环境, 将弹出如图 1.8 所示的“选择默认环境设置”对话框。

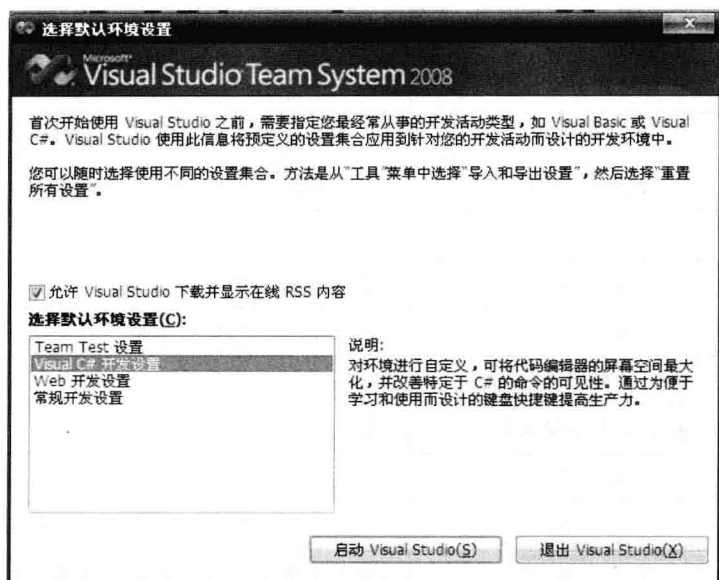


图 1.8 选择默认环境设置

(2) 在图 1.8 所示对话框中选择“Visual C#开发设置”选项，单击“启动 Visual Studio”按钮，即可进入 Visual Studio 2008 开发环境起始页。

1.4.2 菜单栏

菜单栏中集成了所有可用的命令，通过鼠标单击即可执行菜单命令，此外也可以通过按快捷键（如 Alt+相应字母）执行菜单命令。常用的菜单命令及其作用如表 1.2 所示。

表 1.2 常用菜单命令及其作用

菜单项	菜单命令	功 能
文 件	新 建	建立一个新的项目、网站、文件等
	打 开	打开一个已经存在的项目、文件等
	添 加	添加一个项目到当前所编辑的项目中
	关 闭	关闭当前页面
	关闭解决方案	关闭当前解决方案
	保存 Form1.cs	保存项目中的当前窗体
	Form1.cs 另存为	将项目中当前窗体换名或者改变路径保存
	全部保存	将项目中所有文件保存
	导出模板	将当前项目作为模板保存起来，生成.zip 文件
	页面设置	设置打印机及打印属性
	打 印	打印指定内容

续表 1.2

菜单项	菜单命令	功 能
文 件	最近的文件	打开最近操作的文件（例如类文件）
	最近的文档	打开最近操作的文档（例如解决方案）
	退 出	退出集成开发环境
编 辑	撤 销	撤销上一步操作
	重 复	重做上一步所做的修改
	撤销上次全局操作	撤销上一步全局操作
	重复上次全局操作	重做上一步所做的全局修改
	剪 切	将选定内容放入剪贴板，同时删除文档中所选的内容
	复 制	将选定内容放入剪贴板，但不删除文档中所选的内容
	粘 贴	将剪贴板中的内容粘贴到当前光标处
	删 除	删除所选内容
	从数据库删除表	将表从数据库中删除
	全 选	选择当前文档中的全部内容
	查找和替换	在当前窗口文件中查找指定内容，可将查找到的内容替换为指定信息
	转 到	选择定位到“结果”窗格的哪一行
	书 签	显示书签功能菜单
视 图	代 码	显示代码编辑窗口
	设计器	打开设计器窗口
	服务器资源管理器	显示服务器资源管理器窗口
	解决方案资源管理器	显示解决方案资源管理器窗口
	类视图	显示类视图窗口
	代码定义窗口	显示代码定义窗口
	对象浏览器	显示对象浏览器窗口
	错误列表	显示错误列表窗口
	输 出	显示输出窗口
	属性窗口	显示属性窗口
	任务列表	显示任务列表窗口
	工具箱	显示工具箱窗口
	查找结果	显示查找结果
	其他窗口	显示其他窗口（例如命令窗口、起始页等）

续表 1.2

菜单项	菜单命令	功 能
视 图	工具栏	打开工具栏菜单（例如标准工具栏、调试工具栏）
	显示窗格	用于“查询”和“视图设计器”中的显示窗格
	工具箱	显示工具箱
	全屏显示	将当前窗体全屏显示
	向后定位	将控制权移交给下一任务
	向前定位	将控制权移交给上一任务
	属性页	为用户控件显示属性页
项 目	添加 Windows 窗体	添加一个窗体
	添加用户控件	添加一用户控件
	添加组件	添加某个组件
	添加类	添加类文件
	添加新项	添加一个新项到当前所编辑的项目中
	添加现有项	添加一个已存在的项到当前所编辑的项目中
	添加新的分局式系统关系图	为当前项目添加新的分局式系统关系图
	从项目中移除	将当前项目移除
	显示所有文件	在资源管理器中显示当前项目文件下的所有文件
	添加引用	为当前项目添加引用
	添加 Web 引用	为当前项目添加 Web 引用
	设为启动项目	将选定的项目设为启动项
	项目属性	设置项目的属性
	生成解决方案	将项目生成解决方案
生 成	重新生成解决方案	将以前的项目删除，重新生成解决方案
	清理解决方案	清除项目的解决方案
	生成项目	生成项目
	重新生成项目	重新生成项目
	清理项目	清理项目
	发布项目	发布项目
	对项目的代码进行分析	对项目的代码进行分析，检测代码正确性
	批生成	将当前项目成批生成
	配置管理器	打开配置管理器
	窗 口	窗口功能菜单（包括断点、输出、即时）
调 试	启动调试	启动项目并可以调试错误