



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

高职高专计算机系列

C# 程序设计 项目式教程

the C# Programming Language

鄢军霞 ◎ 主编

杨国勋 王燕波 孙琳 ◎ 副主编

王路群 ◎ 主审

- + 国家级骨干校**重点建设**教材
- + **零基础** C# 编程从入门到提高
- + 基于 **Visual C#2010** 的开发环境



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

C# 程序设计 项目式教程

the C# Programming Language

本书通过实际案例，全面阐述了使用Visual C#2010开发应用程序的基本知识。本书分为7章。主要内容包括Visual C#2010的开发环境；C#程序的变量和数据类型等基本语法，C#的语句结构、操作字符串、面向对象中类与对象的基本知识，面向对象的高级应用，基于Windows的程序设计，使用ADO.NET管理数据，使用LINQ访问数据。

免费提供

PPT等教学相关资料



人民邮电出版社
教学服务与资源网
www.ptpedu.com.cn

教材服务热线：010-81055256

反馈/投稿/推荐信箱：315@ptpress.com.cn

人民邮电出版社教学服务与资源网：www.ptpedu.com.cn



ISBN 978-7-115-36130-1



9 787115 361301 >

ISBN 978-7-115-36130-1

定价：35.00 元

封面设计：董志桢



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

C# 程序设计 项目式教程

高职高专计算机系列

the C# Programming Language

鄢军霞 ◎ 主编

杨国勋 王燕波 孙琳 ◎ 副主编

王路群 ◎ 主审

人民邮电出版社

北京

图书在版编目 (C I P) 数据

C#程序设计项目式教程 / 鄢军霞主编. — 北京 : 人民邮电出版社, 2014. 8
工业和信息化人才培养规划教材. 高职高专计算机系列

ISBN 978-7-115-36130-1

I. ①C… II. ①鄢… III. ①C语言—程序设计—高等教育—教材 IV. ①TP312

中国版本图书馆CIP数据核字 (2014) 第130510号

内 容 提 要

本书通过实际的案例, 全面阐述了使用 Visual C#2010 开发应用程序的基本知识。本书分为 7 章, 主要内容包括 Visual C#2010 的开发环境; C#程序的变量和数据类型等基本语法, C#的语句结构、操作字符串、面向对象中类与对象的基本知识, 面向对象的高级应用, 基于 Windows 的程序设计; 使用 ADO.NET 管理数据; 使用 LINQ 访问数据等。

本书适合 C#编程学习的入门者, 也适合有一定 C#基础且想继续深入学习 C#编程的读者, 可以作为高职高专院校学生的学习用书和教师的参考用书。

-
- ◆ 主 编 鄢军霞
副 主 编 杨国勋 王燕波 孙 琳
主 审 王路群
责任编辑 王 威
责任印制 杨林杰
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京圣夫亚美印刷有限公司印刷
- ◆ 开本: 787×1092 1/16
印张: 14.75 2014 年 8 月第 1 版
字数: 377 千字 2014 年 8 月北京第 1 次印刷
-

定价: 35.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316
反盗版热线: (010)81055315

前言 PREFACE

微软为了推行.NET 战略,特别为.NET 平台设计了一种新语言——C#。C#是由 C 和 C++发展而来的一种“简单、高效、面向对象、类型安全”的程序设计语言,综合了 Visual Basic 的高效率和 C++的强大功能。C#已成为.NET 平台的主流语言,它是.NET 的关键语言,是整个.NET 平台的基础。

本书作为全国骨干校计算机及相关专业指定教材,针对全国骨干校软件工程职业学院的特点,以够用为度、深入浅出、重视动手及实际操作技能的培养为目的,适合具有一定计算机基础知识的学生作为教材或自学用书。

本书的读者可以是编程的入门者,甚至可以完全没有编程经验。本书从基本的语法开始,一步一步、深入浅出地介绍应如何使用 C#语言进行各种应用程序的设计。通过学习本书的实例并借助 C#语言的强大功能,读者可以很快从基础学习到如何进行数据库编程设计。同时,本书也适合作为职业院校教师的参考用书。

在内容的安排上,本书以.NET 4.0 和 Visual C# 2010 为基础进行介绍。第1章主要讲解 C#概述和.NET 集成开发环境,以及 C#数据类型、变量和常量、运算符的应用、C#语句结构等内容。第2章介绍操作字符串的相关知识。第3章介绍面向对象的相关概念以及面向对象程序设计中的类的定义,包括字段、属性、方法、构造函数、索引器、静态成员的定义与使用,对象创建等基本知识内容。第4章介绍面向对象思想的三个核心要素,即继承、封装与多态,以及密封类、抽象类、接口、委托与事件等相关知识。第5章介绍 Windows 程序的基本结构、窗体、属性和事件,Windows 程序常用控件以及这些控件的使用方法等内容。第6章介绍在 C#中使用 ADO.NET 管理数据的相关技术,主要包括如何使用命令对象和数据集来访问 SQL Server 数据库数据的技术。第7章介绍在 C#中使用 LINQ 访问数据的相关技术,主要包括 LINQ to DataSet 和 LINQ to SQL 的使用方法,以及如何使用 DataGridView 控件来显示查询数据。

本书深入浅出,从最基本的语法开始讲起,逐步深入到面向对象、Windows 程序设计、数据库高级编程方法。在介绍语法时,本书并没有像一些语法书一样教条而刻板地讲定义,而是试图利用示例代码生动地让读者在实践中体会一个个知识点。通过分析示例代码和书中的讲解,对本书每一章后面的习题进行练习,读者可以很快掌握 C#语言的精髓。主要特点如下。

1. 案例与理论教学紧密结合

为了使读者能够快速理解 Visual C# 2010 中的相关技能并熟练运用,本书的每一章节中都由案例引入开始,导出问题,再讲述所需要使用的理论知识,最后由本章所讲述的理论知识完成本章的案例,让读者在实际的运用中更好地掌握相关理论知识。

2. 合理、有效的组织

本书按照由浅入深的顺序,循序渐进、系统地介绍了 Visual C# 2010 的相关知识和技能。各个章节的编写以实践应用为目标,理论知识的组织紧紧围绕应用技术组织和展

开, 实践的重要性得到体现。

3. 配有全部的程序源文件和电子教案

为方便读者使用, 书中全部实例的源代码及电子教案均免费赠送给读者。读者可以从人民邮电出版社教学服务与资源网免费下载, 网址为 www.ptpedu.com.cn。

本书由王路群担任主审, 第1章、第5章由王燕波编写, 第2章由孙琳编写, 第3章、第4章由鄢军霞编写, 第6章、第7章由杨国勋编写。此外, 姚登峰、涂洪涛、董宁、陈娜、谢日星、李唯、张松慧参加了本书的编写工作, 鄢军霞统编全稿。

本书是作者在多年的教学实践、科学研究以及项目实践的基础上, 参阅了大量的国内外相关教材后, 几经修改而成。

作者的学习成长离不开网络和书籍。在编写此书过程中, 作者更是置身于现实的学术氛围中, 无疑要吸纳与借鉴专家和同行们先进的学术思想、方法, 在此深深地感谢他们给予的启迪和帮助。

由于编者水平有限, 在内容选材和叙述上难免有不当之处, 欢迎广大读者对本书提出批评和建议。E-mail: 826560859@qq.com。

此书得到国家自然科学基金面上项目(61171114)、湖北省教育科学“十二五”规划2012年度立项课题(2012B259)以及北京高等学校青年英才计划项目(YETP1753)资助。

编者

2014年5月

目 录 CONTENTS

第 1 章 C#语法基础 1

1.1 C#和.NET 集成开发环境		1.4.2 变量的类型	20
Visual Studio 2010	1	1.4.3 变量的作用域	20
1.1.1 认识 C#	1	1.4.4 常量的声明和使用	21
1.1.2 安装 Visual Studio 2010	2	1.5 运算符的应用	21
1.1.3 Visual Studio 2010 界面介绍	5	1.5.1 算术运算符	21
1.2 认识 C#程序	11	1.5.2 关系运算符	24
1.2.1 命名空间的定义和使用	11	1.5.3 逻辑运算符	25
1.2.2 指示符 using	11	1.5.4 赋值运算符	26
1.2.3 class 关键字	12	1.5.5 运算符的优先级	26
1.2.4 声明 Main 方法	12	1.5.6 运算符的结合顺序	27
1.2.5 给 C#程序添加说明——注释	12	1.6 C#语句结构	27
1.3 数据类型	13	1.6.1 分支选择结构	27
1.3.1 值类型	13	1.6.2 循环结构	28
1.3.2 引用类型	14	1.6.3 异常处理语句	29
1.3.3 类型转换	17	本章小结	31
1.4 变量和常量	17	习题	31
1.4.1 变量的声明和使用	19		

第 2 章 操作字符串 33

2.1 案例引入	33	2.3 可变字符串	43
2.2 字符串	34	2.3.1 StringBuilder 简单介绍	43
2.2.1 字符串的基本概念	34	2.3.2 StringBuilder 的作用	44
2.2.2 字符串的查找	36	2.4 使用正则表达式	48
2.2.3 字符串的比较	39	2.4.1 正则表达式概述	48
2.2.4 字符串的插入	40	2.4.2 使用正则表达式替换文本	49
2.2.5 字符串的删除	41	2.4.3 使用正则表达式搜索	50
2.2.6 子字符串的获得	42	2.5 回到案例	52
2.2.7 字符串的替换	43	本章小结	53
		习题	54

第3章 面向对象的程序设计 55

3.1 案例引入	55	3.9 构造函数	84
3.2 面向对象概述	55	3.9.1 构造函数概述	84
3.3 类与对象	56	3.9.2 默认构造函数	85
3.3.1 类的定义	56	3.9.3 显式声明的无参构造函数	86
3.3.2 对象的定义	57	3.9.4 构造函数的重载	88
3.4 字段与属性	57	3.9.5 指定初始值设定项	90
3.4.1 字段的定义	57	3.9.6 readonly 修饰符	93
3.4.2 字段的使用	59	3.10 静态成员	93
3.4.3 属性的定义	60	3.10.1 静态字段	94
3.4.4 属性的使用	63	3.10.2 静态属性	94
3.5 方法	65	3.10.3 静态方法	95
3.5.1 方法的定义	65	3.10.4 静态构造函数	96
3.5.2 方法的调用	66	3.10.5 静态类	100
3.6 值类型与引用类型	69	3.11 索引器	101
3.6.1 值类型与引用类型的区别	69	3.11.1 索引器的定义	105
3.6.2 装箱与拆箱	71	3.11.2 索引器的使用	107
3.7 参数的传递	72	3.12 内部类和匿名类	111
3.7.1 按值传递	73	3.12.1 内部类	111
3.7.2 引用传递	74	3.12.2 匿名类	111
3.7.3 ref 引用传递	76	3.13 案例完成	112
3.7.4 out 输出参数传递	77	本章小结	114
3.8 方法的重载	79	习题	114

第4章 面向对象的高级特性 115

4.1 案例引入	115	4.6.4 base 关键字	130
4.2 面向对象的三大特性	116	4.7 密封类	132
4.3 类的继承	116	4.8 抽象类	134
4.4 构造函数的执行	119	4.9 接口	136
4.5 访问修饰符	121	4.10 委托与事件	143
4.5.1 类的可访问性	121	4.10.1 委托	143
4.5.2 类中各成员的可访问性	122	4.10.2 事件	146
4.6 类的多态	126	4.11 案例解决	148
4.6.1 方法的重载	126	本章小结	158
4.6.2 成员的隐藏	126	习题	158
4.6.3 虚方法	128		

第5章 Windows 应用程序 160

5.1	Windows 程序的基本结构	160	5.3.7	ListView 控件	173
5.2	窗体、属性、事件	161	5.3.8	TreeView 控件	175
5.3	常用控件	162	5.4	菜单与上下文菜单	177
5.3.1	RadioButton 控件	162	5.5	工具条	178
5.3.2	CheckBox 控件	164	5.6	状态条	179
5.3.3	Panel 控件	166	5.7	消息框	181
5.3.4	GroupBox 控件	167	5.8	MDI	183
5.3.5	ListBox 控件	168	5.9	窗体跳转	187
5.3.6	ComboBox 控件	172	本章小结		187
			习题		187

第6章 使用 ADO.NET 管理数据 188

6.1	案例引入	188	6.5.2	查询数据	196
6.2	ADO.NET 概述	189	6.5.3	获取表的信息	197
6.3	数据库连接	190	6.6	数据集	199
6.4	命令对象	192	6.6.1	数据集与数据适配器	199
6.4.1	创建命令对象	192	6.6.2	数据集中的数据修改	200
6.4.2	执行 SQL 文本命令	193	6.6.3	添加记录行	202
6.4.3	执行存储过程	194	6.6.4	在 DataSet 中访问多个表	203
6.5	数据读取器	196	6.7	回到案例	206
6.5.1	数据读取器概述	196	本章小结		210
			习题		210

第7章 使用 LINQ 访问数据 211

7.1	案例引入	211	7.5	LINQ to SQL	217
7.2	LINQ 概述	212	7.6	回到案例	223
7.3	LINQ to Objects	212	本章小结		228
7.4	LINQ to DataSet	214	习题		228

【本章学习目标】



本章主要讲解 C#概述和.NET 集成开发环境, 以及 C#数据类型、变量和常量、运算符的应用、C#语句结构等内容。通过本章的学习, 读者应该掌握以下内容:

- 了解 C#语言特点;
- 理解 C#程序的结构;
- 掌握变量和常量、运算符的应用、C#语句结构的使用。

1.1 C#和.NET 集成开发环境 Visual Studio 2010

1.1.1 认识 C#

C# (读作“C sharp”) 是一种编程语言, 它是为生成在 .NET Framework 上运行的各种应用程序而设计的。C#简单、功能强大、类型安全, 而且是面向对象的。C#凭借在许多方面的创新, 在保持 C 语言风格的表现力和雅致特征的同时, 实现了应用程序的快速开发。

C#语法表现力强, 而且简单易学。C#的大括号语法使任何熟悉 C、C++或 Java 的人都可以立即上手。了解上述任何一种语言的开发人员通常在很短的时间内就可以开始使用 C#高效地进行工作。C#的生成过程比 C 和 C++简单, 比 Java 更为灵活, 没有单独的头文件, 也不要按照特定顺序声明方法和类型。C#源文件可以定义任意数量的类、结构、接口和事件。

C#语法简化了 C++的诸多复杂性, 并提供了很多强大的功能, 例如可为 null 的值类型、枚举、委托、lambda 表达式和直接内存访问。这些都是 Java 所不具备的。C#支持泛型方法和类型, 从而提供了更出色的类型安全和性能。C#还提供了迭代器, 允许集合类的实施者定义自定义的迭代行为, 以便容易被客户端代码使用。语言集成查询 (LINQ) 表达式使强类型查询成为了一流的语言构造。

作为一种面向对象的语言, C#支持封装、继承和多态性的概念。所有的变量和方法, 包括 Main 方法 (应用程序的入口点), 都封装在类定义中。类可能直接从一个父类继承, 但它可以实现任意数量的接口。重写父类中的虚方法的各种方法, 要求 override 关键字作为一种

避免意外重定义的方式。在 C# 中，结构类似于一个轻量类。它是一种堆栈分配的类型，可以实现接口，但不支持继承。

在 C# 中，如果必须与其他 Windows 软件（如 COM 对象或本机 Win32 DLL）交互，则可以通过一个称为“互操作”的过程来实现。互操作使 C# 程序几乎能够完成本机 C++ 应用程序可以完成的任何任务。

1.1.2 安装 Visual Studio 2010

Microsoft Visual Studio 2010 是面向 Windows Vista、Office 2010、Web 2.0 的下一代开发工具，是对 Visual Studio 2007 的升级。

在安装 Visual Studio 2010 之前，首先确保 IE 浏览器版本为 6.0 或更高。同时，可安装 Visual Studio 2010 开发环境的计算机配置要求如下所示。

1. 软件要求

Visual Studio 2010 可以安装在以下操作系统上：

- Windows XP (x86) Service Pack 3—除 Starter Edition 之外的所有版本
- Windows Vista (x86 & x64) Service Pack 1—除 Starter Edition 之外的所有版本
- Windows 7 (x86 & x64)
- Windows Server 2003 (x86 & x64) Service Pack 2
- Windows Server 2003 R2 (x86 & x64)
- Windows Server 2008 (x86 & x64) Service Pack 2
- Windows Server 2008 R2 (x64)

支持的体系结构：

- 32 位 (x86)
- 64 位 (x64)

2. 硬件要求

- 配有 1.6 GHz 或更快处理器的计算机
- 1 024 MB 内存
- 3 GB 可用硬盘空间
- 5 400 RPM 硬盘驱动器
- DirectX 9 视频卡，1 280 × 1 024 或更高显示分辨率
- DVD-ROM 驱动器

当开发计算机满足以上条件后就能够安装 Visual Studio 2010。安装 Visual Studio 2010 的过程比较简单。

(1) 将 Visual Studio 2010 安装盘放到光驱中，光盘自动运行后会进入安装程序文件界面。如果光盘不能自动运行，可以双击 setup.exe 可执行文件，如图 1-1 所示。之后安装程序会自动跳转到“Visual Studio 2010 安装程序”界面。

(2) 进入 Visual Studio 2010 界面后可以看到这里有两个安装选项：安装 Microsoft Visual Studio 2010 和检查 Service Release。一般情况下，单击第一项【安装 Microsoft Visual Studio 2010】开始 Visual Studio 2010 的安装，如图 1-2 所示。

cab63.cab	14,103 KB	WinRAR 压缩文件	2010-5-5 4:55
cab64.cab	4,796 KB	WinRAR 压缩文件	2010-5-5 4:54
cab65.cab	5,777 KB	WinRAR 压缩文件	2010-5-5 4:57
cab66.cab	5,526 KB	WinRAR 压缩文件	2010-5-5 4:58
cab67.cab	11,325 KB	WinRAR 压缩文件	2010-5-5 4:58
cab68.cab	22,078 KB	WinRAR 压缩文件	2010-5-5 5:05
htmlite.dll	165 KB	应用程序扩展	2010-3-19 10:51
locdata.ini	1 KB	配置设置	2010-5-5 5:08
readme.htm	2 KB	HTML 文档	2010-5-4 16:19
setup.exe	685 KB	应用程序	2010-5-5 5:11
setup.ini	11 KB	配置设置	2010-5-5 5:08
vs_setup.msi	8,593 KB	Windows Install...	2010-5-5 5:08

图 1-1 找到安装包中的 setup.exe 文件



图 1-2 Visual Studio 2010 安装界面

Visual Studio 2010 的安装程序在进行 Visual Studio 2010 的安装前首先会加载安装组件。这些组件为 Visual Studio 2010 的顺利安装提供了基础保障。用户必须等待安装程序完成加载组件后才能继续后面的安装步骤,如图 1-3 所示。

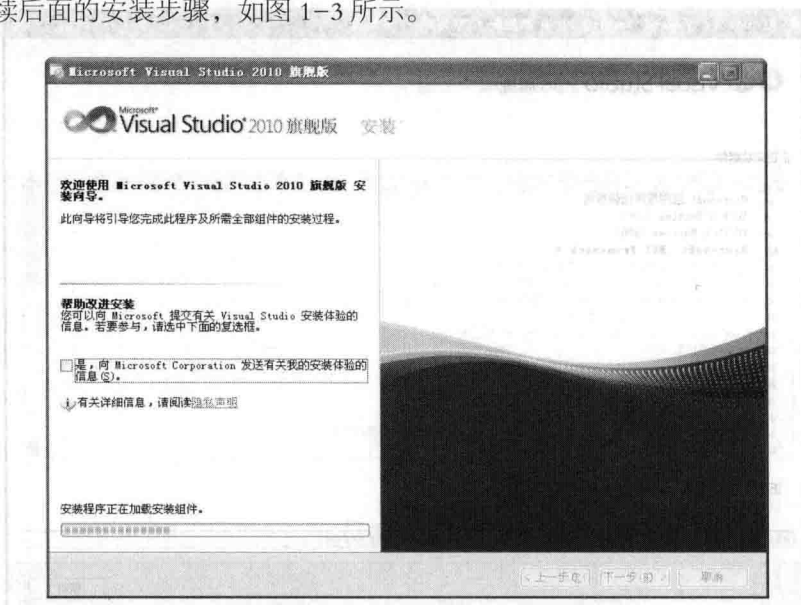


图 1-3 加载安装组件

(3) 在安装组件加载完毕后,单击【下一步】按钮开始 Visual Studio 2010 的安装。在这个界面里,用户可以指定 Visual Studio 2010 的安装路径,如图 1-4 所示。

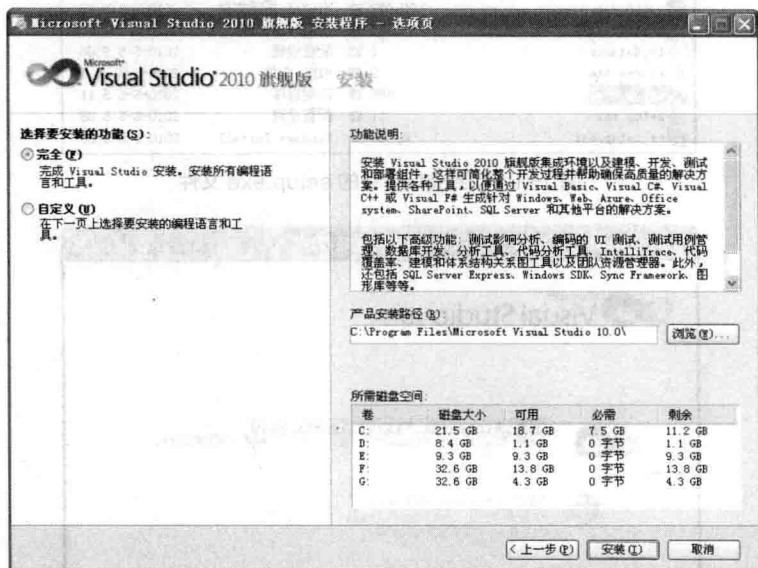


图 1-4 选择 Visual Studio 2010 安装路径

在这个界面里除了可以指定安装路径外,还可以选择相应的安装功能,用户可以选择“完全”和“自定义”。选择“完全”将安装 Visual Studio 2010 的所有组件。如果用户只需要安装几个组件,可以选择“自定义”进行组件的选择安装。

(4) 在用户做好相应的选择后,单击【安装】按钮就开始 Visual Studio 2010 的安装,如图 1-5 所示。

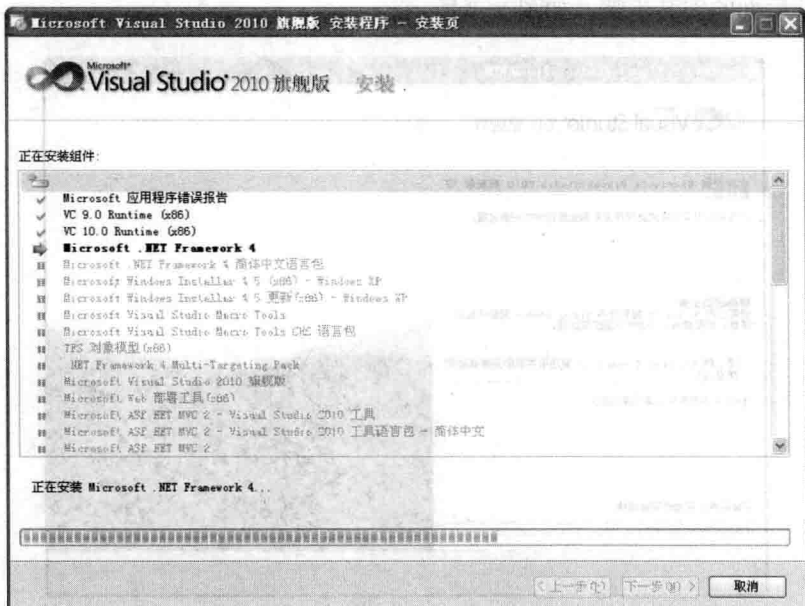


图 1-5 Visual Studio 2010 的安装

图 1-5 中的安装界面显示了安装列表和当前安装进度。安装完毕后就会出现安装成功界面,说明已经在本地计算机中成功安装了 Visual Studio 2010。

1.1.3 Visual Studio 2010 界面介绍

1. 主窗口

在 Visual Studio 2010 安装完成后,就可以进行 .NET 应用程序的开发。Visual Studio 2010 极大地提高了开发人员对 .NET 应用程序的开发效率。首先需要熟悉一下 Visual Studio 2010 开发环境。先来看看 Visual Studio 2010 的主窗口,如图 1-6 所示。

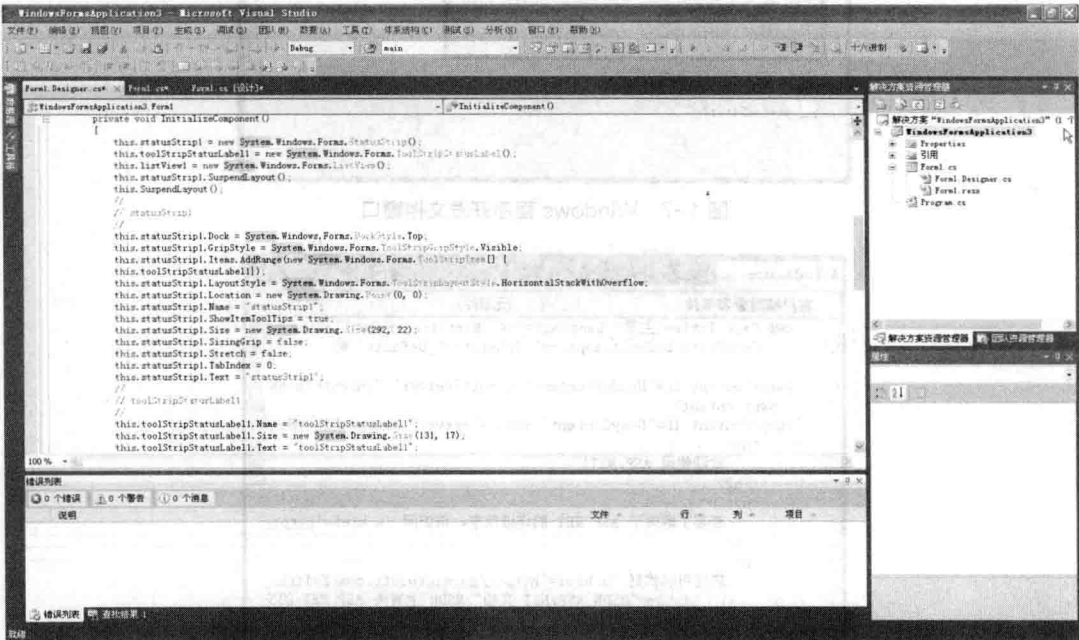


图 1-6 Visual Studio 2010 主窗口

如图 1-6 所示, Visual Studio 2010 主窗口包括多个子窗口。左侧是数据源和工具箱,当前处于折叠状态;中间是文档窗口,用于应用程序代码的编写和样式控制;下方是错误列表窗口,用于显示错误信息;右侧是资源管理器窗口和属性窗口,用于呈现解决方案,以及页面及控件的相应属性。

2. 文档窗口

文档窗口用于代码的编写和样式设计。当用户开发的是基于 Web 的 ASP.NET 应用程序时,文档窗口是以 Web 的形式呈现给用户的,而代码视图则是以 HTML 代码的形式呈现给用户的。而如果用户开发的是基于 Windows 的应用程序,则文档窗口将会呈现应用程序的窗口或代码,如图 1-7 和图 1-8 所示。

当开发人员在进行不同的应用程序开发时,文档窗口也会自动呈现为不同的样式以方便开发人员使用。例如在 ASP.NET 应用程序中,其文档窗口包括三部分,如图 1-9 所示。

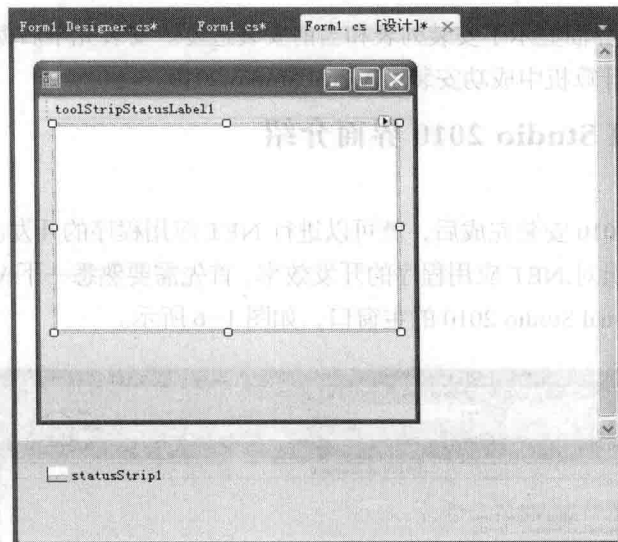


图 1-7 Windows 程序开发文档窗口

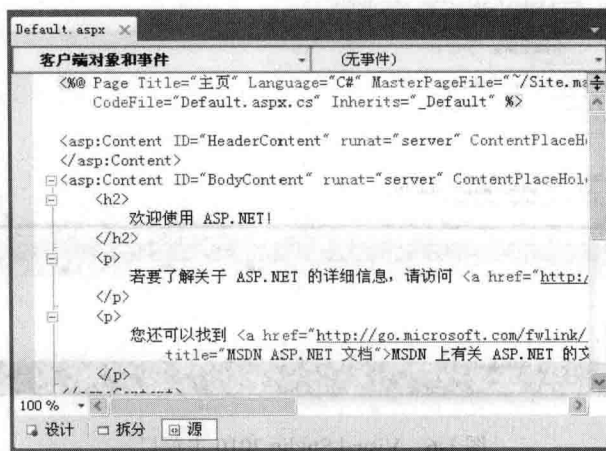


图 1-8 Web 程序开发文档窗口



图 1-9 文档窗口

开发人员可以通过使用这三部分进行高效开发，其具体功能如下。

- 页面标签：当进行多个页面开发时，会呈现多个页面标签。当开发人员需要在不同的页面之间切换时，可以通过页面标签进行快速切换。
- 视图栏：用户可以通过视图栏进行视图的切换。Visual Studio 2010 提供“设计”、“拆分”和“源代码”三种视图，开发人员可以选择不同的视图进行页面样式控制和代码的开发。
- 标签导航栏：标签导航栏能够进行不同标签的选择。可以通过标签导航栏进行标签或标签内容的选择。

开发人员可以灵活运用文档主窗口进行高效的应用程序开发。Visual Studio 2010 的视图栏窗口提供了拆分窗口，拆分窗口允许开发人员进行页面样式开发和代码编写。

3. 工具箱

Visual Studio 2010 主窗口的左侧为开发人员提供了工具箱，工具箱中包含了 Visual Studio 2010 对 .NET 应用程序所支持的控件。对于不同的应用程序开发而言，在工具箱中所显示出来的工具也不同。工具箱是 Visual Studio 2010 中的基本窗口，开发人员可以使用工具箱中的控件进行应用程序开发，如图 1-10 和图 1-11 所示。

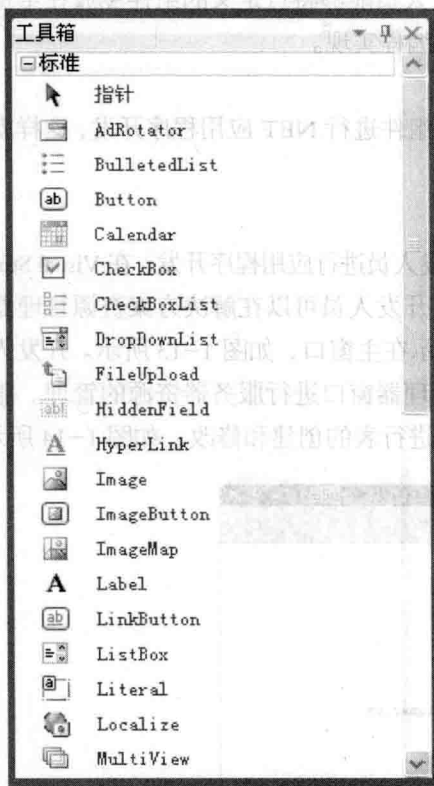


图 1-10 工具箱



图 1-11 选择类别

正如图 1-11 所示，系统默认为开发人员提供了数十种控件用于系统的开发，用户也可以添加工具箱选项卡进行自定义组件的存放。Visual Studio 2010 为开发人员提供了不同类别的

控件。这些控件被归为不同的类别，开发人员可以按照需求进行相应类别的控件的使用。开发人员还能够在工具箱中添加现有的控件。右击工具箱空白区域，在下拉菜单中选择【选择项】选项，系统会弹出窗口用于开发人员对自定义控件的添加，如图 1-12 所示。



图 1-12 添加自定义组件

组件添加完成后就会显示在工具箱中，开发人员能够将自定义的组件曳放在主窗口中进行应用程序中相应功能的开发，而无需通过复杂编程实现。

注意：

开发人员能够使用他人已经开发好的自定义组件进行.NET 应用程序开发，这样就无需通过编程实现重复的功能。

4. 解决方案资源管理器

在 Visual Studio 2010 的开发中，为了方便开发人员进行应用程序开发，在 Visual Studio 2010 主窗口的右侧会显示一个解决方案资源管理器。开发人员可以在解决方案资源管理器中进行相应文件的选择，双击后相应文件的代码就会显示在主窗口，如图 1-13 所示。开发人员还可以单击解决方案资源管理器下方的服务器资源管理器窗口进行服务器资源的管理。服务器资源管理器还允许开发人员在 Visual Studio 2010 中进行表的创建和修改，如图 1-14 所示。



图 1-13 解决方案资源管理器