

# Microsoft Visual C#.NET 程序设计

John Sharp      著  
Jon Jagger  
朱峰奎      译

北京大学出版社

· 北 京 ·

## 内 容 简 介

C#是在 Microsoft.NET 开发平台中占有极重要地位的一套程序语言,它不是 C/C++或是 Visual Basic 的任何一个新版本,而是整合二者优点及特色于一身的新一代编程语言。本书共分 6 大部分 32 章详细介绍了 Visual Studio.NET 和 Microsoft Visual C#基础知识、C#语言基础、组件创建、Windows 应用程序创建、数据管理方法以及 Web 应用程序生成等 C#应用内容。随书所附光盘包含各章所用的不同阶段的文件,极大地方便了 C#爱好者与开发者的学习与阅读,是编程爱好者学习 C#编程的启蒙与应用读本。

著作权登记号: 01-2001-4250

### 图书在版编目(CIP)数据

Microsoft Visual C#.NET 程序设计/(美)夏普(Sharp,J.), (美)吉格(Jagger,J.)著;朱峰奎译. -北京:北京大学出版社, 2002.4

ISBN 7-301-05552-8

I .M... II.①夏... ②吉... ③朱... III.C 语言-程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2002)第 016460 号

书 名: Microsoft Visual C#.NET 程序设计

著作责任者: John Sharp Jon Jagger

译 者: 朱峰奎

责任编辑: 赵乐静

标准书号: ISBN 7-301-05552-8/TP·0656

出 版 者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电 话: 发行部 62754140 62765127 编辑室 62765126 邮购部 62752015

电子信箱: [macrowin@263.net.cn](mailto:macrowin@263.net.cn)

排 版 者: 北京东方人华科技有限公司

印 刷 者: 北京大学印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×980 毫米 16 开本 27.25 印张 710 千字

2002 年 4 月第 1 版 2002 年 4 月第 1 次印刷

定 价: 52.00 元(含光盘)

# 导 言

Microsoft Visual C# .NET是一种功能强大而使用简单的语言，主要用于创建Microsoft .NET平台的应用程序。它继承了C++和Microsoft Visual Basic的很多优秀特征，剔除了一些不一致和错误，因而更为简洁和逻辑化。C# 还包含很多创新，从而加快了应用程序的开发速度，特别是在与Microsoft Visual Studio.NET关联使用时更是如此。

本书的目的是帮助读者学习使用Visual Studio.NET和.NET框架来进行C#编程的基本原理。书中将学习C#语言的特征，然后生成运行在Microsoft Windows操作系统下的应用程序。本书将使用C#生成Windows窗体应用程序、操作XML、访问SQL Server、控制Windows服务、开发ASP.NET应用程序以及生成并使用Web服务。完成本书的学习后，您将对C#有彻底了解。

## 最小系统需求

要完成本书的练习，您需要具备如下硬件和软件配置：

- Microsoft Windows XP Home 或 Professional, Microsoft Windows NT 4.0 Server, Microsoft Windows 2000 Professional 或 Microsoft Windows 2000 Server
- Microsoft Visual Studio.NET Professional 或 Enterprise
- 如果运行 Microsoft Visual Studio.NET Enterprise, 需要安装并配置 MSDE; 如果运行 Microsoft Visual Studio.NET Professional Edition, 需要访问 Microsoft SQL Server 2000 或更高版本
- 访问运行在 MSDE 或 SQL Server 下的 Northwind Traders 数据库
- 安装并运行 Microsoft IIS 5.0
- Pentium III 450 MHz 或兼容处理器(建议 733 MHz)
- 3GB 可用硬盘空间
- 64MB RAM(建议 128MB)
- Super VGA 显示器(800\*600 或更高分辨率), 256 色
- CD-ROM
- Microsoft 鼠标或兼容指示设备

## 找到您的最佳学习起点

本书可以在很大程度上帮助您学会使用技巧。如果您是编程初学者或学习其他程序语言(如 C、C++、Sun Microsystem 的 Java 或 Microsoft Visual Basic 6)的读者,现在希望学习 C#,请查看下表找到最佳学习起点。

对象	学习步骤
面向对象编程初学者	<ol style="list-style-type: none"> <li>1 根据下面的“安装和使用练习文件”部分安装练习文件。</li> <li>2 依次学习第 1~17 章的各章内容。</li> <li>3 根据经验级别和兴趣完成第 18~32 章的学习。</li> </ol>
C#初学者	<ol style="list-style-type: none"> <li>1 根据下面的“安装和使用练习文件”部分安装练习文件,略读第 1~5 章对 C# 和 Visual Studio.NET 有个总的了解,然后专注于第 6~17 章。</li> <li>2 根据经验级别和兴趣完成第 18~26 章的学习。</li> </ol>
学习 C#的 C、C++或 Java 程序员	<ol style="list-style-type: none"> <li>1 根据下面的“安装和使用练习文件”部分安装练习文件。</li> <li>2 略读前 7 章以对 C#和 Visual Studio.NET 有个总的认识,然后专著于第 8~17 章。</li> <li>3 关于生成 Windows 应用程序和使用数据库的内容,阅读 18~26 章。</li> <li>4 关于生成 Web 应用程序和 Web 服务的内容,阅读第 6 部分。</li> </ol>
学习 C#的 Visual Basic 6 程序员	<ol style="list-style-type: none"> <li>1 根据下面的“安装和使用练习文件”部分安装练习文件。</li> <li>2 依次学习第 1~17 章的各章内容。</li> <li>3 关于生成 Windows 应用程序的内容,阅读第 18~23 章。</li> <li>4 关于访问数据库的内容,阅读第 24~26 章。</li> <li>5 关于创建 Web 应用程序和 Web 服务的内容,阅读第 27~32 章。</li> </ol>
完成练习后参考本书	根据下面的“安装和使用练习文件”部分安装练习文件。

## 安装和使用练习文件

随书光盘包含进行本书练习时需要用到的练习文件。例如,在学习 if 语句时,将使用到一个写好的项目 Selection。通过使用练习文件,不必将时间浪费在创建与练习无关的项目上,因此可以专注于学习以 Visual C#.NET 面向对象编程。练习文件和循序渐进的指导寓学于用,可以轻松有效地获取和记忆新技术。

### 安装练习文件

根据以下步骤将练习文件安装到本地硬盘,以便在练习中使用。

#### 1 取出光盘并放入光驱。



**注意** 光盘应该是自动运行的,如果没有,双击桌面上的【我的电脑】图标,双击光驱所在盘符图标,然后双击 StartCD.exe。

- 2 单击【安装练习文件】，出现一个消息框。
- 3 单击【确定】按钮。出现选择选项窗口，推荐选项已经选中，为了有效使用本书练习文件，建议接受这些默认选中的选项。
- 4 安装完成后，从光驱中取出光盘并将其放回光盘套中。本地硬盘中即创建了一个名为\Visual C# Step by Step 的文件夹，其中包括所有的练习文件。

## 使用练习文件

本书的每一章都解释了何时应该以及如何使用其练习文件。使用练习文件时，书中将指导如何打开要使用的文件。书中章节是模拟真实编程项目的情况编写的，在现实工作中可以很轻松地运用从本书中学到的技术。

下面是练习文件中所有 Visual C#项目的列表。

章 名	项 目	描 述
第 1 章	TextHello	该项目开始进入学习过程，一步步介绍了一个显示基于文本的欢迎程序的创建过程。
	WinFormHello	该项目使用 Windows 窗体在窗口中显示欢迎界面。
第 2 章	PrimitiveDataTypes	该项目描述如何声明各种基本类型的变量、如何为这些变量赋值以及如何如何在窗口中显示变量的值
	MathsOperators	该程序介绍算术运算符(+、-、*、/与%)
第 3 章	Methods	在该项目中，将再次查看上一个练习的代码，并查看它是如何使用方法建构的
	DailyRate	该项目介绍如何编写方法(手动或使用向导)、执行方法以及使用 Visual Studio 调试器逐语句调试方法
第 4 章	Selection	该项目显示如何使用级联 if 语句比较两个日期
	switchStatement	该简单程序使用 switch 语句将字符转换为对应的 XML 表示
第 5 章	Iteration	该项目显示每个不同迭代语句的代码片段以及它们所生成的输出
	whileStatement	该项目使用一条 while 语句来一次读取一个源文件的一行内容并在 Windows 文本框中显示每一行
	doStatement	该项目使用一条 do 语句将一个数字转换为其字符串表示
第 6 章	MathsOperators	该项目再次查看第 2 章的 MathsOperators 项目并引发不同的未处理异常来导致程序运行失败。然后使用 try 和 catch 关键字加强应用程序的健壮性，且程序运行不再失败
第 7 章	Classes	该项目包括定义用户类的基础、使用公有构造函数、方法和私有字段。还包括使用关键字 new 创建类实例以及静态方法和字段
第 8 章	Parameters	该程序研究值参数和引用参数的不同，说明如何使用关键字 ref 和 out
第 9 章	StructsAndEnums	该项目使用一个枚举表示一年中的四季，然后使用一个结构体表示一个日期

续表

章 名	项 目	描 述
第 10 章	Aggregates	该项目使用 ArrayList 集合类模拟纸牌游戏的收牌、洗牌与发牌
第 11 章	ParamsArrays	该项目显示如何使用关键字 params 创建一个单个的接受任何 int 传递参数的方法, 该方法然后查找并返回其中的最小值
第 12 章	CSharp	该项目使用接口和类的继承来模拟阅读 C#源文件并将其内容分类为不同种类的标号 (标识符、关键字和运算符等)。作为一个使用实例, 也从主接口派生类来在富文本框中以颜色语法显示标号
第 13 章	UsingStatement	该项目再次查看第 12 章的一部分代码并指出它不是异常安全的, 显示如何使用 using 语句使代码处于异常安全模式
第 14 章	Properties	该项目显示了一个简单的 Windows 应用程序, 该程序使用几个属性来连续显示主窗口的大小
第 15 章	Indexers	该项目使用两个索引器: 一个在给出名字时查找此人的电话号码, 另一个在给出电话号码时查找人名
第 16 章	Delegates	该项目使用代理回调以数字格式显示时间。然后使用事件简化代码
第 17 章	Operators	该项目在前一个项目基础上生成, 使用 3 个包含用户定义的运算符的结构体 Hour、Minut 和 Second, 然后使用转换运算符简化代码
第 18 章	BellRingers	该项目是一个 Windows 窗体应用程序, 用来说明基本 Windows 窗体控件
第 19 章	BellRingers	该项目是第 18 章所创建的项目的扩展, 但在用户界面中添加了下拉列表和弹出菜单
第 20 章	CustomerDetails	该项目以客户信息为例说明如何验证用户输入
第 21 章	SplitterDemo	该项目显示如何使用 Splitter 控件, 并显示不同控件的 Dock 属性如何使用
	BellRingers	该项目说明如何使用 TreeView 和 ListView 控件
第 22 章	MDIDemo	该项目是一个显示如何生成 MDI 应用程序的文本编辑器, 并显示如何使用通用对话框控件
第 23 章	LoginControl	该项目是一个 Windows 控件, 提供了以一般方式登录系统的逻辑
	LoginTest	该项目是 LoginControl 项目的测试使用
	SubclassedControls	该项目使用继承创建 CheckBox 类的一个特殊版本 ToggleButton
	ToggleButtonTest	该项目是 ToggleButton 的测试使用
第 24 章	ListProducts	该项目实现访问 Northwind Traders 数据库的数据连接
	DisplayProducts	该项目使用 ListProducts 项目中创建的数据连接来连接到 Northwind Traders 数据库, 然后使用 SqlDataAdapter 和 DataSet 控件检索并更正 Products 表中的信息
	ReportOrders	该项目显示如何使用 Microsoft ADO.NET 代码而不是数据连接和数据控件访问数据库, 该应用程序从 Orders 表中检索信息

续表

章 名	项 目	描 述
第 25 章	DataBindingDemo	该项目显示如何使用简单数据绑定来绑定控件属性到不同数据源中的单个值
	ComplexBindingDemo	该项目显示如何使用复杂数据绑定从数据源中显示一列值
	ManageTerritories	该项目说明如何使用数据绑定和非连接数据集检索并维护 Northwind Traders 数据库中 Territories 表中的信息
第 26 章	RecordTimesheets	该项目定义一个 XML 架构并使用该架构将用户输入转换为 XML
	AnalyzeTimesheets	该项目验证并分析使用在 RecordTimesheets 项目中定义的 XML 架构的 XML 文件中的数据
第 27 章	EmployeeInfo	该项目是一个简单的 ASP.NET 应用程序,该程序介绍 Web 窗体和 Server 控件
第 28 章	EmployeeInfo	该项目是第 27 章 EmployeeInfo 项目的扩展版本,显示如何在一个 ASP.NET Web 应用程序中验证用户输入
第 29 章	CustomerInfo	该项目显示如何使用基于窗体的安全性鉴别用户。还讲述了如何从一个 ASP.NET Web 窗体使用 ADO.NET,显示如何以可升级的方式操作数据库
第 30 章	SupplierInfo	该项目显示如何在数据网格中排序数据以及如何在 ASP.NET Web 应用程序中缓存数据
	ProductInfo	该项目显示如何使用列模板来在一个 DataGrid 控件中定制列外观
	ProductCategories	该项目显示了如何在 ASP.NET 应用程序的两个 Web 窗体间导航以及如何从一个 Web 窗体向另一个 Web 窗体传递信息
第 31 章	NorthwindServices	该项目实现一个 XML Web 服务,提供通过 Internet 对 Northwind Traders 数据库的 Products 表中数据的远程访问
第 32 章	OrderCost	该项目显示如何创建 XML Web 服务使用,它是一个 Windows 应用程序,并显示如何同步调用第 31 章创建的 XML Web 服务
	ProductInfo	该项目也是一个 XML Web 服务使用,显示如何异步调用一个 Web 服务

除这些项目之外,还有一些项目有练习的解决方案,每个项目的解决方案存储于附书光盘中,并在各章对应的文件夹中以 Complete 标出。

### 卸载练习文件

根据以下步骤从计算机中删除练习文件,这些步骤是针对 Windows XP Professional 操作系统而写的,如果使用不同版本的 Windows,参看 Windows 帮助文档以卸载程序。

- 1 选择【开始】|【控制面板】命令。
- 2 单击【添加/删除程序】图标。
- 3 从【已安装的程序】列表中选择 Microsoft Visual C#.NET Step by Step 选项。

- 4 单击【更改/删除】按钮，打开【确认文件删除】对话框。
- 5 单击【是】按钮删除练习文件。
- 6 单击【关闭】按钮关闭【添加或删除程序】对话框。
- 7 关闭【控制面板】。

# 目 录

<b>第 1 章 C#</b> .....	1
1.1 使用 Visual Studio.NET 环境开始编程.....	1
1.2 开始写程序.....	3
1.3 使用命名空间.....	6
1.4 使用 XML 和注释创建文档.....	8
1.5 创建 Windows 窗体应用程序.....	11
<b>第 2 章 变量、运算符及表达式</b> .....	18
2.1 语句.....	18
2.2 使用标识符.....	18
2.3 标识关键字.....	19
2.4 使用变量.....	19
2.5 基本数据类型.....	21
2.6 设置算术运算符.....	24
<b>第 3 章 方法和作用域</b> .....	29
3.1 声明方法.....	29
3.2 调用方法.....	32
3.3 作用域.....	33
3.4 写并调用方法.....	35
<b>第 4 章 使用判定语句</b> .....	41
4.1 声明布尔变量.....	41
4.2 使用布尔运算符.....	41
4.3 执行 if 语句.....	43
4.4 使用 switch 语句.....	48
<b>第 5 章 使用迭代</b> .....	52
5.1 使用复合赋值运算符.....	52

5.2	递增和递减变量.....	53
5.3	写 while 语句.....	54
5.4	写 for 语句.....	55
5.5	写 do 语句.....	56
<b>第 6 章</b>	<b>管理错误和异常.....</b>	<b>65</b>
6.1	处理错误.....	65
6.2	测试代码和捕获异常.....	66
6.3	使用复合 catch 处理器.....	67
6.4	写常规 catch 处理器.....	67
6.5	使用 checked 和 unchecked 整数运算.....	70
6.6	Throwing 异常.....	74
6.7	写最后一块代码.....	78
<b>第 7 章</b>	<b>创建并管理类和对象.....</b>	<b>80</b>
7.1	什么是分类.....	80
7.2	什么是封装.....	80
7.3	控制可访问性.....	81
7.4	使用构造函数和 new 关键字.....	82
7.5	理解静态方法和数据.....	89
<b>第 8 章</b>	<b>值和引用.....</b>	<b>94</b>
8.1	复制 int 变量和类.....	94
8.2	栈和堆.....	100
8.3	什么是 System.Object.....	101
8.4	装箱.....	102
8.5	出箱.....	103
<b>第 9 章</b>	<b>创建枚举和结构体类型.....</b>	<b>105</b>
9.1	枚举.....	105
9.2	结构体类型.....	108
<b>第 10 章</b>	<b>数组与集合.....</b>	<b>118</b>
10.1	数组.....	118

---

10.2 什么是集合类 .....	124
<b>第 11 章 参数数组 .....</b>	<b>134</b>
11.1 创建重载方法 .....	134
11.2 使用数组参数 .....	134
<b>第 12 章 继承 .....</b>	<b>144</b>
12.1 什么是继承 .....	144
12.2 理解核心语法 .....	144
12.3 创建接口 .....	149
12.4 使用多重接口 .....	160
<b>第 13 章 使用垃圾搜集器和资源管理站 .....</b>	<b>163</b>
13.1 垃圾搜集器 .....	163
13.2 资源管理 .....	167
<b>第 14 章 运用属性来存取属性 .....</b>	<b>175</b>
14.1 比较字段和方法 .....	175
14.2 什么是属性 .....	176
14.3 理解属性的限制 .....	179
14.4 使用静态属性 .....	180
14.5 声明接口属性 .....	181
<b>第 15 章 使用索引 .....</b>	<b>187</b>
15.1 什么是索引 .....	187
15.2 接口索引 .....	192
15.3 在 Window 应用程序中使用索引 .....	193
<b>第 16 章 代理和事件 .....</b>	<b>198</b>
16.1 使用代理声明及实例 .....	198
16.2 使用事件加快公告生成 .....	206
<b>第 17 章 运算符重载 .....</b>	<b>212</b>
17.1 运算符 .....	212
17.2 声明转换运算符 .....	220

---

<b>第 18 章 Windows 窗体介绍</b> .....	224
18.1 创建应用程序 .....	224
18.2 向窗体添加控件 .....	229
18.3 在 Windows 窗体中处理事件 .....	236
<b>第 19 章 菜单</b> .....	239
19.1 菜单制作标准和样式 .....	239
19.2 添加菜单并处理菜单事件 .....	240
19.3 弹出菜单 .....	245
<b>第 20 章 验证</b> .....	248
20.1 验证数据 .....	248
20.2 示例——Customer Maintenance .....	249
<b>第 21 章 使用复合控件</b> .....	257
21.1 资源管理器界面 .....	257
21.2 分隔栏窗口、插接控件和面板 .....	258
21.3 导航数据的控件 .....	262
<b>第 22 章 使用 MDI、窗口和对话框</b> .....	275
22.1 什么是多文档界面 .....	275
22.2 创建对话框 .....	280
22.3 使用通用对话控件 .....	283
<b>第 23 章 创建 GUI 组件</b> .....	286
23.1 使用用户控件 .....	286
23.2 指定控件使用 Subclassing .....	294
<b>第 24 章 使用数据库</b> .....	300
24.1 使用 ADO.NET 数据库 .....	300
24.2 使用 Northwind Traders 数据库 .....	300
<b>第 25 章 使用数据绑定和数据集</b> .....	314
25.1 Windows 窗体控件和数据绑定 .....	314

---

25.2 非连接数据集.....	320
<b>第 26 章 XML 解析.....</b>	<b>327</b>
26.1 为什么使用 XML.....	327
26.2 雇员工作表系统.....	330
26.3 创建 XML 架构.....	330
<b>第 27 章 ASP.NET 简介.....</b>	<b>341</b>
27.1 将 Internet 理解为基本结构.....	341
27.2 使用 ASP.NET 创建 Web 应用程序.....	343
<b>第 28 章 验证控件.....</b>	<b>358</b>
28.1 比较服务器端和客户端验证.....	358
28.2 实现客户端验证.....	359
<b>第 29 章 通过 Web 窗体访问数据.....</b>	<b>364</b>
29.1 使用 Web 窗体的 DataGrid 控件.....	364
29.2 查询数据.....	368
29.3 编辑数据.....	374
<b>第 30 章 生成 ASP.NET 应用程序.....</b>	<b>380</b>
30.1 DataGrid 控件的其他特征.....	380
30.2 在窗体间导航.....	387
<b>第 31 章 生成 XML Web 服务.....</b>	<b>393</b>
31.1 什么是 XML Web 服务.....	393
31.2 生成 ProductService Web 服务.....	397
<b>第 32 章 使用 Web 服务.....</b>	<b>407</b>
32.1 Web 服务、客户端和代理.....	407
32.2 异步执行 Web 方法.....	412

# 第 1 章 C#

## 本章知识要点:

- ✓ 启动 Microsoft Visual Studio.NET
- ✓ 使用 Visual Studio.NET 编程环境
- ✓ 创建 C# 控制台应用程序
- ✓ 创建 C# Windows 窗体应用程序

Microsoft Visual C#.NET 是 Microsoft 公司发布的强大的面向组件的语言。C#在 Microsoft.NET 框架结构中起着重要的作用,相当于 C 在 UNIX 发展过程中所起的作用。C#使用大括号,因此,熟悉诸如 C、C++或 Java 等大括号语言的用户会发现 C#的语法相当相似。

本书第 1 部分将学习 C#的基础,包括如何声明变量及如何使用运算符(如+和-)创建值,如何写方法并向方法传递参数,还将学习如何使用选择语句(如 if)和条件语句(如 while)。最后将可以知道 C#是如何使用异常来以一种容易使用的方式处理错误的。这些内容组成了 C#的核心,有了这样坚固的基础后,在第 2 部分到第 6 部分还将学习更高级的特性。

## 1.1 使用 Visual Studio.NET 环境开始编程

Visual Studio.NET 是一个包含很多工具的编程环境,它包括创建大的或小的 C#项目(甚至可以创建与其他语言的模块无缝连接的项目)需要用到的所有功能。第一个练习将启动 Visual Studio.NET 编程环境并学习如何创建一个控制台应用程序。

### 在 Visual Studio.NET 中创建控制台应用程序

- 1 在【开始】菜单中选择【程序】| Microsoft Visual.NET | Microsoft Visual.NET 命令,启动 Visual Studio.NET,如图 1.1 所示。
- 2 选择【文件】|【新建】|【项目】命令,打开【新建项目】对话框,如图 1.2 所示。此对话框允许用户使用指定不同应用程序类型(如 Windows 应用程序、类库和控制台应用程序)的模板创建新项目。
- 3 在【项目类型】面板中单击【Visual C#项目】图标。
- 4 在【模板】面板中单击【控制台应用程序】图标(可能需要滚动面板查看此项)。
- 5 在【位置】文本框中输入“C:\\Visual step by step\\Chapter 1”。

- 6 在【名称】文本框中输入“TextHello”。
- 7 单击【确定】按钮，打开新项目。

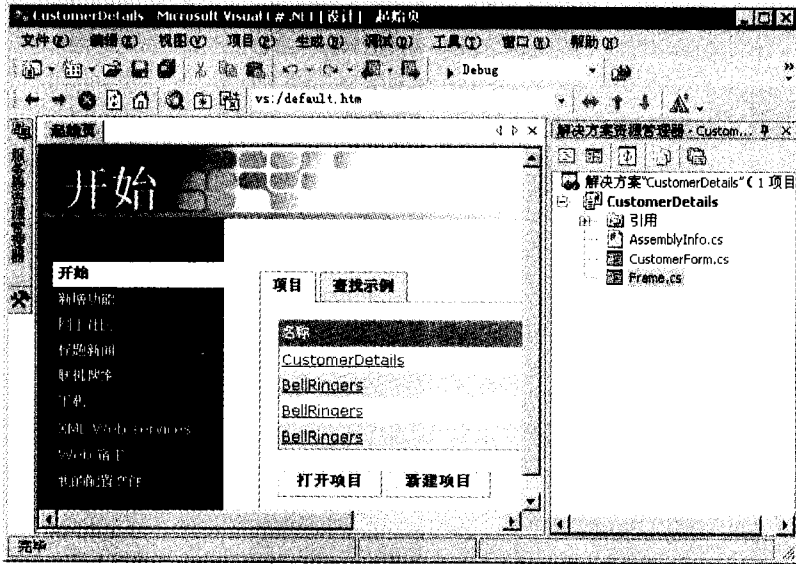


图 1.1

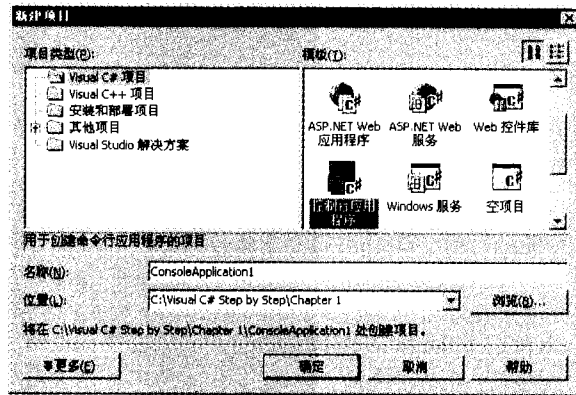


图 1.2

**提示** 如果指定的文件夹不存在，Visual Studio.NET 将创建一个。

通过菜单栏可以访问编程环境中将要用到的命令，像在所有基于窗体的程序中一样，也可以使用键盘或鼠标访问这些命令。工具栏位于菜单栏的下方，包括了最常用命令的快捷按钮。代码窗口用于显示源文件。在一个多文件项目中，每个源文件都有自己的标有源文件名的标签。单击标签可将相应的源文件提到代码窗口的最前方。【解决方案资源管理器】显示与该项目有关的文件的文件

名称。在【解决方案资源管理器】中双击文件名可以将该源文件提到代码窗口的最前方。输出面板列出了通过编译命令生成的诊断信息。

写代码前首先检查【解决方案资源管理器】中列出的 Visual Studio.NET 创建的作为项目组成部分的文件。

- **TextHello.sln** 这是顶级解决方案文件，每个应用程序都必须有一个。每个解决方案文件包含一个或几个项目文件。在文件系统中，解决方案文件的扩展名是.sln。在【解决方案资源管理器】中显示为一个易读的名字，如在此例中为 Solution ‘TextHello’。
- **TextHello.csproj** 这是一个 C#项目文件。每个项目文件包含一个或多个源文件。一个项目的源文件必须以相同的程序语言编写。此类文件在【解决方案资源管理器】中只显示项目名，它的文件扩展名为.csproj。
- **Class1.cs** 这是一个 C#源文件，将在此文件中写代码。它包含了一些 Visual Studio.NET 自动提供的代码。
- **AssemblyInfo.cs** 这也是一个 C#源文件。可以使用此文件向程序添加属性，如作者名和程序编写日期等。也有其他高级属性，可用来改变程序运行方式。
- **App.ico** 这是一个与应用程序相关的图标。所有包含用户界面的 Windows 应用程序都有一个在应用程序运行时最小化到任务栏的图标。

如果使用 Windows 资源管理器浏览\\Visual C# Step by Step\Chapter 1 文件夹指定项目路径，将会发现一个名为 TextHello 的文件夹(根据 Windows 设置不同，此文件夹可能被隐藏)。此文件夹中包含【解决方案资源管理器】中列出的文件。

## 1.2 开始写程序

文件 Class1.cs 定义了一个名为 Class1 的类，此类包含一个名为 Main 的方法。所有的方法都必须在一个类中定义。Main 方法比较特殊，它指明了程序的入口，必须是一个静态方法(方法将在第 3 章讨论，静态方法在第 7 章讨论，Main 方法在第 9 章讨论)。



**重点** C#区分大小写，Main 的第一个字母必须大写。

下面的练习是编写一段代码在控制台上显示“Hello World”；生成并执行该 Hello World 应用程序；学习命名空间是如何划分代码元素的，以及如何使用 XML 注释代码，然后以 HTML 文档查看 XML 注释。

### 使用 IntelliSense 技术写代码


1 在 Class1.cs 的 Main 方法中删除 TODO 注释(以//开头的一行)。Main 方法如下：

```
static void Main(string[] args)
```

```
{  
}
```

- 2 在左括号后加入一空白行并输入“Console”，Main方法如下：


```
static void Main(string[] args)  
{  
    Console  
}
```

 **提示** Console 是一个内置类，其中包含在屏幕上显示信息与从键盘输入信息的方法。

- 3 在 Console 后输入一个点号，出现一个 IntelliSense 列表，其中包括 Console 类的所有方法、属性及字段。

- 4 向下滚动列表，选中 WriteLine，然后按 Enter 键。也可以滚动列表并双击 WriteLine。列表关闭，而且.WriteLine方法添加到源文件中。现在 Main方法如下：

```
static void Main(string[] args)  
{  
    Console.WriteLine  
}
```

 **提示** 当 IntelliSense 列表出现后，按 W 键可以选中以 W 开头的第 1 个 Console 成员。

- 5 输入一个左括号“(”，出现一个 IntelliSense 提示，该提示显示 WriteLine 方法的参数。事实上，WriteLine 是一个重载方法，也就是说 Console 包含多个名为 WriteLine 的方法(重载方法将在第 3 章讨论)。现在 Main方法如下：

```
static void Main(string[] args)  
{  
    Console.WriteLine(  
}
```

- 6 输入右括号和一个分号，现在 Main方法如下：

```
static void Main(string[] args)  
{  
    Console.WriteLine();  
}
```

- 7 在括号中输入字符串“Hello World”，现在 Main方法如下：

```
static void Main(string[] args)  
{  
    Console.WriteLine("Hello World");  
}
```