



程序员指南丛书

Visual Basic.NET 编程基础

- 使用 Visual Basic.NET,能够在代码中方便地创建自己的组件或配置.NET 框架类中的复杂组件。
- 通过大量实例,深入浅出地介绍了 Visual Basic.NET 的使用与编程,通俗易懂、易于理解掌握。
- 适用于各个层次读者,不仅可以快速了解 Visual Basic.NET 新增功能,还可以进一步提高编程技术。

张文 等编著



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



Visual Basic.NET 编程基础

张 文 等 编 著

清华大学出版社

(京)新登字 158 号

内 容 简 介

Visual Basic.NET 是微软公司推出的新一代可视化面向对象开发工具。本书通过大量实例,深入浅出地介绍了 Visual Basic.NET 的使用与编程。全书共 8 章,主要包括:Visual Basic.NET 开发环境和菜单体系、Visual Basic.NET 语言、Windows 窗体编程基础、管理和使用 Visual Basic.NET 对象、窗体应用程序的构造、窗体界面设计以及 Windows 服务应用程序等。

本书内容全面深入,适合于初中级读者、大中专院校师生、企业技术开发人员学习参考,也适合于各类培训班学员学习 Visual Basic 程序设计技术。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: Visual Basic.NET 编程基础
作 者: 张文 等
出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)
<http://www.tup.tsinghua.edu.cn>
责任编辑: 胡先福
印 刷 者: 世界知识印刷厂
发 行 者: 新华书店总店北京发行所
开 本: 787 × 1092 1/16 印张: 18.25 字数: 437 千字
版 次: 2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷
书 号: ISBN 7-302-04963-7/TP · 2795
印 数: 0001 ~ 6000
定 价: 26.80 元

前 言

Visual Basic.NET 是 Visual Studio.NET 家族的一个重要成员, Visual Studio.NET 家族还包括 Visual C#.NET、Visual C++ .NET 等开发工具。根据习惯, Visual Studio.NET 和 Visual Basic.NET 又被称为 Visual Studio 7.0 和 Visual Basic 7.0。

Visual Basic.NET 是微软公司推出的 Visual Basic 语言的新一代产品, 其设计目的是为了快速而简便地开发包括 Web 服务和 ASP.NET Web 应用程序在内的 .NET 框架应用程序。使用 Visual Basic.NET, 能够在代码中方便地创建自己的组件或配置 .NET 框架类中的复杂组件。使用非可视化组件能够在自己的应用程序中方便地使用消息队列、事件日志、性能计数等资源。

Visual Basic.NET 中引入了很多更新和改进的语言性能, 例如继承、接口和重载, 这使其成为一种强大的面向对象的编程语言。其他新的语言特性还包括自由线程和结构化异常处理等。Visual Basic.NET 能与 .NET 框架及通用语言运行库完全整合, 因而能够提供多语言互用、冗码收集、增强的安全性和改进的版本支持。

Windows 窗体是 Visual Basic.NET 的主要编程对象, 它是一种新型的面向对象的框架, 实现了 .NET 框架 Windows 窗体和 Windows 窗体控件, 并提供了强健的 Windows 应用程序开发结构。

本书通过大量实例, 深入浅出地介绍了 Visual Basic.NET 的使用与编程。全书共 8 章, 主要内容包括: Visual Basic.NET 开发环境和菜单体系、Visual Basic.NET 语言、Windows 窗体编程基础、管理和使用 Visual Basic.NET 对象、窗体应用程序的构造、窗体界面设计以及 Windows 服务应用程序等。

本书通俗易懂、易于理解掌握, 对 Visual Basic.NET 的几乎所有方面都进行了介绍, 并且假定读者以前从未用 Visual Basic 编过程序; 对于使用 Visual Basic 以前版本或其他结构化语言编过程序的读者来说, 通过本书不仅可以快速了解 Visual Basic.NET 新增功能, 还可以进一步提高编程技术。

本书并非只是知识点的简单罗列, 而是通过实例向读者全面讲解 Visual Basic.NET 的语言知识点和编程技巧, 这使得读者能够掌握并灵活运用这些知识点, 迅速掌握这门编程语言。

除封面署名外, 参与本书编写、校对等工作的还有吴清萍、安浦、方克志、于丽闽、李国文、周银昆、金立言、张明翔、罗忠、李平梅、刘瑞芳、陈建平、王世雄、江宏、朱胜武、段爱华等。由于水平所限, 不足和纰漏之处在所难免, 恳请广大读者批评指正。

作者

2001 年 10 月

目 录

第 1 章 掌握 Visual Basic.NET 开发环境	1
1.1 Visual Studio.NET 的安装需求	1
1.1.1 硬件需求	1
1.1.2 软件需求	2
1.2 Visual Basic.NET 的新特性	2
1.2.1 Web 开发的新特性	2
1.2.2 数据开发的新特性	3
1.2.3 编程语言的新特性	3
1.2.4 Windows 窗体和控件的新特性	3
1.2.5 项目的新特性	3
1.2.6 调试的新特性	4
1.2.7 组件和组件创建的新特性	4
1.2.8 国际化应用程序的新特性	4
1.3 Visual Studio.NET 集成开发环境简介	4
1.3.1 先进的窗口管理技术	5
1.3.2 强大的编辑功能	6
1.4 使用 Visual Studio.NET 集成开发环境	7
1.4.1 Visual Studio.NET 起始页	7
1.4.2 方案管理器	12
1.4.3 Class View	15
1.4.4 属性窗口	16
1.4.5 工具箱	17
1.4.6 编辑器	29
1.4.7 服务器管理器	30
1.4.8 对象浏览器	32
1.4.9 任务清单窗口	36
1.4.10 宏管理器	38
1.4.11 命令窗口	41
1.4.12 安装工具	42
本章小结	42

第 2 章 Visual Basic.NET 的菜单体系	43
2.1 File 菜单	43
2.2 Edit 菜单	46
2.3 View 菜单	47
2.4 Project 菜单	48
2.5 Build 菜单	52
2.6 Debug 菜单	53
2.7 Tools 菜单	54
2.8 Window 菜单	58
2.9 Help 菜单	59
本章小结	60
第 3 章 掌握 Visual Basic.NET 语言	61
3.1 Visual Basic.NET 与 .NET 框架	61
3.1.1 部件	61
3.1.2 引用和 Imports 语句	62
3.1.3 名称空间	63
3.1.4 标志	64
3.2 Visual Basic.NET 的新特性	65
3.2.1 新增语言特性	66
3.2.2 改进的语言特性	67
3.3 程序结构和编码惯例	78
3.3.1 Visual Basic.NET 程序结构	79
3.3.2 “你好,世界!”	80
3.3.3 Visual Basic.NET 命名惯例	81
3.3.4 断句和组句	82
3.4 声明元素	82
3.4.1 声明元素名称	82
3.4.2 声明元素的特征	83
3.4.3 生命期	84
3.4.4 作用域	85
3.4.5 声明元素的引用	87
3.4.6 名称隐藏	88
3.5 数据类型	89
3.5.1 数据类型声明	90
3.5.2 标识符类型字符	90
3.5.3 值类型字符	91
3.5.4 基本数据类型	92

3.5.5	复合数据类型	95
3.5.6	数据类型实现	95
3.6	类型转换	97
3.6.1	扩展转换和收缩转换	97
3.6.2	隐式转换和显示转换	98
3.6.3	转换中的值变化	100
3.6.4	字符串和其他类型间的转换	100
3.6.5	数组转换	101
3.6.6	无类型编程	102
3.7	变量	104
3.7.1	变量声明	104
3.7.2	对象变量	105
3.8	运算符和表达式	109
3.8.1	算术运算	109
3.8.2	位运算	111
3.8.3	比较运算	111
3.8.4	连接运算	113
3.8.5	逻辑运算	114
3.9	数组	115
3.9.1	使用数组	115
3.9.2	声明数组变量	116
3.9.3	数组的高级特性	117
3.9.4	数组的数组	118
3.9.5	数组和集合	119
3.10	常量和枚举	119
3.10.1	常量声明	119
3.10.2	常量和值数据类型	120
3.10.3	使用枚举	122
3.11	结构	123
3.11.1	结构声明	124
3.11.2	结构变量	124
3.11.3	结构和其他编程元素	125
3.11.4	结构嵌套	126
3.11.5	结构和类	126
3.12	语句	127
3.12.1	赋值语句	127
3.12.2	声明语句	128
3.12.3	可执行语句	129

3.13 过程	129
3.13.1 过程和结构化代码	129
3.13.2 Sub 过程	130
3.13.3 函数过程	131
3.13.4 属性过程	133
3.13.5 过程参数	134
3.13.6 可选参数	136
3.13.7 根据位置和名称传递参数	137
3.13.8 参数数组	138
3.13.9 递归过程	138
3.14 流程控制	139
3.14.1 选择结构	139
3.14.2 循环结构	142
3.14.3 嵌套控制结构	146
本章小结	146
第4章 Windows 窗体编程基础	147
4.1 Windows 窗体	147
4.1.1 Windows 窗体概述	147
4.1.2 窗体代码结构	149
4.2 事件驱动	149
4.3 Visual Basic .NET 对象	151
4.3.1 类和对象	151
4.3.2 对象成员	153
4.4 创建一个 Windows 窗体应用程序	155
本章小结	158
第5章 管理和使用 Visual Basic .NET 对象	159
5.1 创建和使用对象	159
5.1.1 设置和获取属性	159
5.1.2 通过方法执行动作	160
5.1.3 执行多个动作	161
5.1.4 操作窗体对象	162
5.1.5 使用 New 关键字	162
5.1.6 资源管理	163
5.1.7 将对象传递给过程	164
5.2 管理对象组	166
5.2.1 创建对象数组	166

5.2.2	创建对象集合	168
5.3	在运行时获取类信息	170
5.3.1	确定对象类型	170
5.3.2	使用字符串名调用属性或方法	171
5.4	定制类	172
5.4.1	类声明	172
5.4.2	定义字段和属性	173
5.4.3	定义方法	175
5.4.4	定义事件	175
5.5	深入理解类	177
5.5.1	类与标准模块	177
5.5.2	对象的创建和销毁	177
5.5.3	共享成员	181
5.5.4	类成员多态性	183
	本章小结	185
第6章	构造 Windows 窗体	186
6.1	使用窗体设计器	186
6.1.1	创建窗体	186
6.1.2	设置窗体尺寸	187
6.1.3	设置窗体位置	188
6.1.4	设置窗体边框风格	189
6.1.5	添加和调整控件	190
6.2	窗体中的事件处理	195
6.2.1	事件和 Delegate	195
6.2.2	事件处理函数	196
6.2.3	创建事件处理函数	196
6.2.4	创建默认事件处理函数	199
6.2.5	在运行时创建事件处理函数	199
6.2.6	使用单个事件处理函数响应多个事件	200
6.3	使用常用控件	200
6.3.1	标签控件	200
6.3.2	图片框控件	202
6.3.3	按钮控件	202
6.3.4	复选框控件	203
6.3.5	单选按钮控件	204
6.3.6	文本框控件	204
6.3.7	列表框控件	207

6.3.8	组合框控件	209
6.3.9	编组框控件	210
6.3.10	滚动条控件	211
6.3.11	计时器控件	212
6.4	创建不同形式的窗体	212
6.4.1	创建隐藏窗体	212
6.4.2	创建顶层窗体	213
6.4.3	创建透明窗体	213
6.4.4	启动屏幕	214
6.4.5	模态和非模态窗体	214
	本章小结	215
第7章	设计窗体界面	216
7.1	界面设计基础	216
7.1.1	设计原则	217
7.1.2	Windows 界面规则	218
7.1.3	界面布局原则	218
7.1.4	用户辅助模型	221
7.1.5	选择界面样式	221
7.2	菜单的设计和使用	223
7.2.1	添加常规菜单	223
7.2.2	设计和使用常规菜单	224
7.2.3	使用快捷菜单	230
7.2.4	菜单应用实例	233
7.3	工具栏的设计和使用	236
7.3.1	设计工具栏	236
7.3.2	使用工具栏	237
7.4	状态栏的设计和使用	237
7.4.1	设计状态栏	238
7.4.2	使用状态栏	238
7.5	设计对话框窗体	239
7.5.1	创建对话框	240
7.5.2	获取对话框数据	240
7.5.3	关闭对话框	243
7.6	创建资源管理器风格的界面	244
7.7	设计 MDI 风格的界面	246
7.7.1	创建 MDI 父窗体	246
7.7.2	创建 MDI 子窗体	247

7.7.3	确定活动子窗体	248
7.7.4	向活动子窗体中发送数据	249
7.7.5	排列子窗体	250
7.7.6	保存子窗体的信息	250
7.8	窗体程序设计实例	251
7.8.1	创建程序框架	252
7.8.2	编辑子窗体代码	253
7.8.3	编辑主窗体代码	257
	本章小结	261
第 8 章	Windows 服务应用程序	262
8.1	Windows 服务概述	262
8.1.1	服务应用程序的独特之处	262
8.1.2	服务生命期	263
8.2	创建 Windows 服务应用程序	263
8.2.1	Windows 服务应用程序的结构	264
8.2.2	创建 Windows 服务应用程序的步骤	265
8.2.3	实现服务功能	266
8.2.4	为服务应用程序添加安装程序	267
8.2.5	指定服务的安全上下文	268
8.2.6	安装和卸载服务	268
8.2.7	指定服务的启动方式	269
8.2.8	手动启动服务	269
8.3	调试 Windows 服务	270
8.3.1	调试 Windows 服务的步骤	270
8.3.2	记录服务信息	272
8.3.3	定制事件日志	273
8.3.4	删除定制事件日志	274
8.4	Windows 服务示例	274
	本章小结	277

第 1 章 掌握 Visual Basic.NET 开发环境

Visual Basic.NET 是 Visual Studio.NET 家族的一个成员,该家族包括 Visual C#.NET、Visual Basic.NET、Visual C++ .NET 等开发工具。根据习惯,Visual Studio.NET 和 Visual Basic.NET 又被称为 Visual Studio 7.0 和 Visual Basic 7.0。

Visual Studio.NET 主要用于开发企业规模的 Web 应用程序,以及高性能的桌面应用程序,它为 .NET 框架(.NET Framework)提供了全面支持。.NET 框架由通用语言运行库(Common Language Runtime)、统一编程类和 ASP.NET(Active Server Pages.NET)组成,这些组件主要用于创建 Web 应用程序和 Web 服务。

Visual Studio.NET 为 XML 提供了完全的支持,例如引入了 XML Designer 以简化 XML 编辑和创建等。在 Visual Studio.NET 中,还能使用 Visual Basic、Visual C# 创建和使用 Web 服务。Visual Studio.NET 通过模板为应用程序提供了结构化控制,企业模板可以降低软件设计开发小组的开发难度和成本。

Visual Studio.NET 为 Visual Basic.NET、Visual C++ .NET 和 Visual C#.NET 等提供了统一的集成开发环境(IDE, Integrated Development Environment),其中集成了许多可视化辅助工具,能够大大简化程序的开发和提高编程效率。从界面风格上讲,Visual Studio.NET 属于 XP 系列,而且,相对于 Office XP,Visual Studio.NET 在开发环境上与先前版本有着更大的不同。因此,要使用 Visual Basic.NET 进行开发,首先必须掌握 Visual Studio.NET 开发环境的使用。本章将向读者介绍 Visual Studio.NET 的集成开发环境。

本章要点:

- ❖ Visual Studio.NET 的安装需求
- ❖ Visual Basic.NET 的新特性
- ❖ Visual Studio.NET 集成开发环境的使用

1.1 Visual Studio.NET 的安装需求

由于 Visual Studio.NET 本身的功能非常强大,因此对软硬件要求都较高。

1.1.1 硬件需求

Visual Studio.NET 不能安装在映射驱动器上,而必须在本地驱动器上进行安装。安装的具体硬件需求如下:

- CPU

CPU 至少应为 Pentium II 450 MHz, 推荐使用 Pentium III 600 MHz。

- 内存

如果操作系统为 Windows 2000 Professional, 内存至少应为 96 MB, 推荐使用 128 MB; 如果操作系统为 Windows 2000 Server, 内存至少应为 192 MB, 推荐使用 256 MB。

- 硬盘空间

系统驱动器上至少应有 500 MB 空间, 而安装的目标驱动器上至少应有 2.5 GB 空间。

- CD-ROM 驱动器

除非能直接从网络下载安装, 或硬盘中存在安装原文件, 否则必须具有 CD-ROM 驱动器。

- 显示器

显示器至少应为 800 × 600 256 色, 推荐使用 1024 × 768 16 位增强色。

1.1.2 软件需求

安装 Visual Studio.NET 的具体软件需求如下:

- 操作系统

Windows 2000、Windows XP 和 Windows NT 4.0 及以上版本。

- 其他组件

Windows NT 4.0 Service Pack 6a、Windows 2000 Service Pack 2、Microsoft Windows Installer 2.0、Microsoft Windows Management Infrastructure、Microsoft FrontPage 2000 Web Extensions Client、Microsoft FrontPage 2000 Server Extensions Service 1.2、Microsoft Internet Explorer 6.0、Microsoft Data Access Components 2.7、Microsoft Jet 4.0 Service Pack 3 和 Microsoft .NET Framework。

1.2 Visual Basic.NET 的新特性

在 Visual Studio.NET 中, Visual Basic.NET 和 Visual C# 都具备 RAD(快速应用程序开发, Rapid-Application Development) 支持, 例如项目模板、设计器和其他开发环境特性, 这两种语言都使用 .NET 框架基类。Visual Basic.NET 是 Visual Basic 的最新版本, 其中引入了很多新特性。

1.2.1 Web 开发的新特性

Visual Basic.NET 的一个重要新特性就是能创建 Web 应用程序, 即运行于 Web 服务器上的 Visual Basic 应用程序。

Web 应用程序中最主要的两个开发主题为 Web 窗体和 Web 服务。

- Web 窗体技术允许用户迅速而方便地创建 ASP.NET Web 应用程序的用户界面。Web 窗体页面是对现有 Web 开发工具的变革性进步,它兼有速度和 RAD 开发环境的强大功能。Web 窗体可以输出于任何浏览器或移动设备,并且能自动提交正确的、与浏览器(风格、布局等)兼容的 HTML。
- Web 服务是能通过 Internet 协议,调用其他组件或应用程序的组件。Web 服务允许使用标准协议(例如 HTTP)进行数据交换,而且能通过 XML 消息穿过防火墙移动数据。Web 服务不依赖于某种特定组件技术或对象调用规范,因此无论运行于什么操作系统上,无论使用什么编程语言和组件模型编写的程序,都能访问 Web 服务。

1.2.2 数据开发的新特性

ADO.NET 是 ADO(ActiveX 数据对象——ActiveX Data Objects)的革命性进步,它为 Web 应用程序开发提供了标准编程模型。ADO.NET 是根据离线模型设计的,因此能够使用与数据源隔离的数据快照。

1.2.3 编程语言的新特性

Visual Basic.NET 是 Visual Basic 语言的新一代产品,其设计目的是为了快速而简便地开发包括 Web 服务和 ASP.NET Web 应用程序在内的 .NET 框架应用程序。

Visual Basic.NET 中引入了很多更新和改进的性能,例如继承、接口和重载,这使其成为一种强大的面向对象的编程语言。其他新的语言特性包括自由线程和结构化异常处理等。Visual Basic.NET 能与 .NET 框架和通用语言运行库完全整合,因而能提供多语言互用、冗码收集、增强的安全性和改进的版本支持。

1.2.4 Windows 窗体和控件的新特性

Windows 窗体是一种新型的面向对象的框架,它实现了 .NET 框架 Windows 窗体和 Windows 窗体控件,并提供了强有力的 Windows 应用程序开发结构。

1.2.5 项目的新特性

项目是 Visual Basic.NET 代码的构造块,已完成的应用程序中可能包含多个项目。以下是 Visual Basic.NET 与 Visual Basic 6.0 不同之处:ASP.NET Web 服务项目、ASP.NET Web 应用程序项目、Windows 服务项目、控制台应用程序项目、Web 控件库项目等。

1.2.6 调试的新特性

Visual Studio .NET 中的所有语言使用同样的调试器,它同时具有旧版本的 Visual C++ 和 Visual Basic 调试器的功能,并在很多地方进行了改进,包括:对 Visual Basic、Visual C++、Visual C#、C++ 管理扩展、脚本和 SQL 等语言的交叉调试;Microsoft .NET 通用语言运行库和 Win32 应用程序的调试;对主机或远程主机运行程序的附加调试;多个程序的同时调试(例如,单个 Visual Studio 解决方案中运行的多个程序,或附加于已运行的附加程序);Visual C++ 动态错误检查;缓冲区安全检查等。

在 Visual Basic .NET 中, Debug.Print 方法已被 Debug.Write、Debug.Writelf、Debug.WriteLine 和 Debug.WriteLineIf 所替代。

1.2.7 组件和组件创建的新特性

使用 Visual Basic .NET 能够在代码中方便地创建自己的组件,或配置 .NET 框架类中的复杂组件。使用非可视化组件,能够在自己的应用程序中方便地使用消息队列、事件日志、性能计数等资源。

.NET 框架类极大地扩展了 Visual Basic 的组件编程性能。虽然基类有限,但用户可以使用很大的复杂组件库来配置自己的应用程序,或派生自己的高质量组件。

1.2.8 国际化应用程序的新特性

Visual Basic .NET 为创建国际化应用程序提供了很大支持,这包括两个方面:国际化,设计能用于不同语言的应用程序;本地化,将资源翻译为指定语言。

1.3 Visual Studio .NET 集成开发环境简介

Visual Studio .NET 为 Visual C++ .NET、Visual Basic .NET 和 Visual C# 等提供统一的集成开发环境,从而为开发者提供了很大方便。Visual Studio .NET 的启动界面如图 1-1 所示。

由图 1-1 可见,Visual Studio .NET 的集成开发环境有很多新改进,例如在开发环境中直接显示网页。当首次打开 Visual Studio .NET 时,其中将显示默认的 Visual Studio .NET 起始页(Start Page)。如果 Web 浏览器被打开,则同时还会出现 Web 工具栏,其功能与 IE 中的相同。

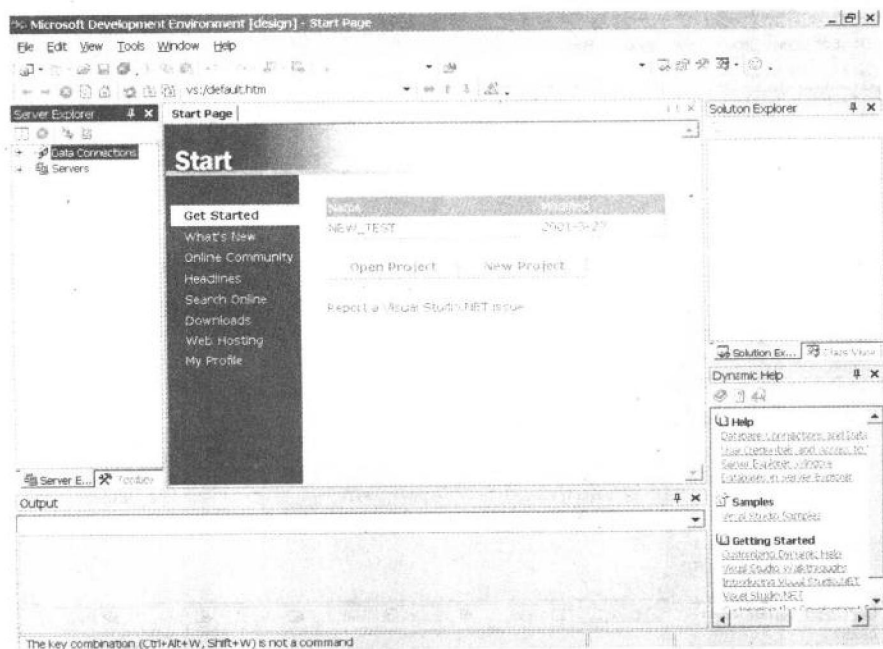


图 1-1 Visual Studio.NET 的启动界面

1.3.1 先进的窗口管理技术

Visual Studio.NET 中采用了更有效的窗口管理策略,例如文档标签化、自动隐藏等,从而大大提高了浏览效率,节省了屏幕空间。图 1-2 所示为 Visual Studio.NET 的典型窗口布局,其中包括集成开发环境中的所有主要窗口。

1. 自动隐藏

在窗口标题栏处单击鼠标右键,在弹出的快捷菜单中选择 Auto Hide 命令,或使目标窗口成为活动窗口,并选择 Window | Auto Hide 命令,即可实现自动隐藏。自动隐藏使得窗口在失去焦点时自动最小化,并将其图标排列在集成开发环境边缘,从而节省宝贵的屏幕空间,增加编辑器的可视面积。

2. 标签化文档

标签化文档功能会自动为文档窗口设置标签,当窗口失去焦点时其标签仍然会显示于原窗口的边缘。例如,默认情况下 Class View、Resource View、宏管理器和方案管理器使用同一窗口,而通过切换底部标签,即可在同一窗口中查看不同视图中的内容。当用户在编辑器或设计器中编辑多个文档时,虽然一次只能显示一个文档中的内容,但是通过顶部或底部标签可以方便地浏览其他文档中的内容,参见图 1-2。与以前版本中必须通过 Window 菜单中

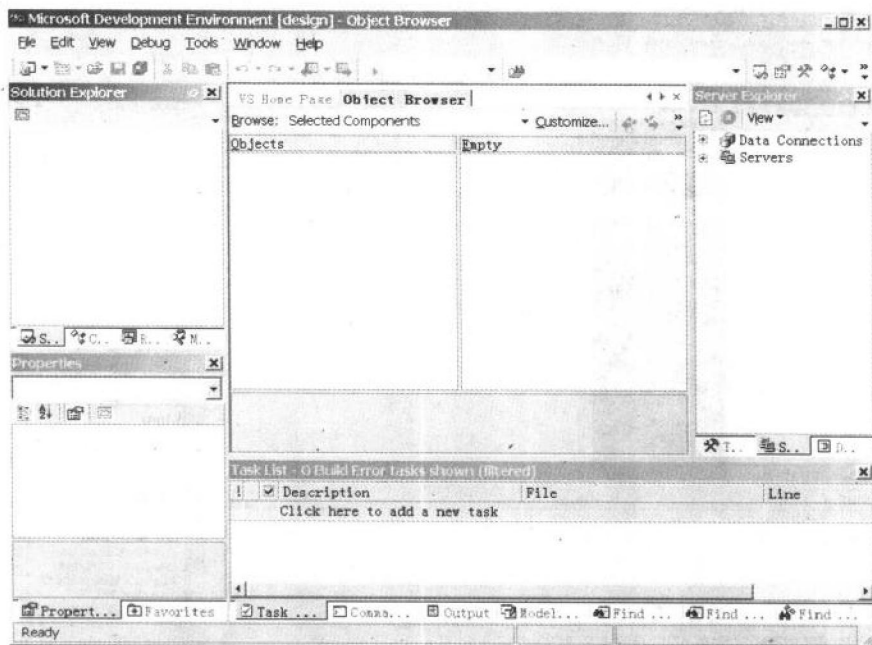


图 1-2 Visual Studio.NET 集成开发环境中的主要窗口

的窗口列表选择相比较,其提供的方便显而易见,尤其是在打开的文档数较多时。

1.3.2 强大的编辑功能

Visual Studio.NET 的编辑工具有了进一步增强:

1. 编辑代码

Visual Studio.NET 的所有产品使用统一的代码编辑器和检索/替换工具,在编辑器中能够实现字回绕、增量搜寻、代码纲要、文本缩进、行编号和书签等增强功能,用户能够通过 Edit 菜单中的命令使用这些功能。使用前进/后退导航,能够在文档的被选内容历史中进行切换。

2. 剪贴板环(Clipboard Ring)

剪贴板环中能够储存系统剪贴板中被拷贝或剪切的最后 12 个条目。使用 Ctrl + Shift + V 快捷键,或通过工具框中的剪贴板标签,用户能够查看剪贴板环中的内容。Ctrl + Shift + V 键将一个条目从剪贴板环中粘贴到文件中的插入点,并自动将被粘贴的条目设置为被选状态。如果再次按下 Ctrl + Shift + V 键,将粘贴下一条目,因此重复按下 Ctrl + Shift + V 键,能够在所有条目中循环。也可以在工具框中的剪贴板标签和文档间通过拖曳交换文本。