



Professional VB.NET 2003

VB.NET 高级编程

(第3版)

(美) Bill Evjen 等著
Billy Hollis 翻译
杨 浩 审校
杨 智



清华大学出版社

VB.NET 高级编程

(第 3 版)

(美)	Bill Evjen	等著
	Billy Hollis	
	杨 浩	翻译
	杨 智	审校

清华大学出版社

北 京

Bill Evjen, Billy Hollis, et al.

Professional VB.NET 2003

EISBN: 0-7645-5992-3

Copyright © 2004 by John Wiley & Sons, Inc.

All Rights Reserved. Authorized translation from the English language edition published by John Wiley & Sons, Inc.

本书中文简体字版由 John Wiley & Sons, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2004-6636

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

VB.NET 高级编程(第3版)/(美)艾文铮(Evjen, B.)(美)赫利斯(Hollis, B.)著; 杨浩翻译. —北京: 清华大学出版社, 2005. 6

书名原文: Professional VB.NET 2003

ISBN 7-302-10599-5

I. V… II. ①艾… ②赫… ③杨… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 016649 号

出版者: 清华大学出版社

<http://www.tup.com.cn>

社总机: 010-62770175

组稿编辑: 曹 康

封面设计: 孔祥丰

印刷者: 清华大学印刷厂

发行者: 新华书店总店北京发行所

开 本: 185×260 印张: 55 字数: 1407 千字

版 次: 2005 年 6 月第 1 版 2005 年 6 月第 1 次印刷

书 号: ISBN 7-302-10599-5/TP·7189

印 数: 1~4000

定 价: 99.00 元

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

文稿编辑: 王 黎

版式设计: 康 博

装 订 者: 三河市李旗庄少明装订厂

译者序

Visual Basic .NET 是世界上使用得最广泛的编程语言。随着 .NET 应用的增多, VB 的重要性也日益凸现。而新推出的 Visual Studio .NET 2003 则将 VB 推向了新的高点, 极大地提高了开发人员的效率, 为公司和企业带来了新的商机。Microsoft 公开宣称, 当生产率是最重要的因素时, VB.NET 是开发应用程序的最佳语言。VB.NET 2003 为广大程序员提供了更为强大、便捷、易用的编程工具。

为了便于读者更全面、深入地了解 VB.NET 2003 的核心功能和新增特性, 本书首先详细解释了 .NET 性能, 随后逐一介绍了公共语言运行库、OOP、命名空间、ADO.NET、Windows 窗体和 Web 窗体、数据绑定、线程、服务, 并特别讲述了 VB.NET 2003 中新增加的针对移动设备的应用程序开发。每章均针对 VB.NET 2003 的新功能进行了详细的阐述和介绍。

全书内容全面, 翔实, 并提供大量的程序示例, 便于读者上机练习, 巩固所学到的编程技术。另外本书的合作站点还提供了源代码下载, 以方便读者实践。

读者朋友在阅读本书时, 宜循序渐进, 按章节顺序进行学习, 以便于系统地了解全书内容, 为您的职业生涯奠定良好的基础。您也可直接选读特定章节, 快速了解相关主题的内容。

由于受时间和译者自身水平的限制, 翻译过程中难免出现错误和疏漏, 敬请大家多多批评指正, 反馈信息请发致 fwkbook@tup.tsinghua.edu.cn 信箱。

译者

前 言

2002 年, Visual Basic 进行了自发布以来最大的变革, 引入了 Visual Basic .NET(VB.NET)。十几年来, VB 一直没有进行大的修订, 但 .NET 已经远远超出了这种修订, 它改变了软件开发的几乎所有方面。从集成 Internet 功能到创建面向对象的架构, VB.NET 向传统的 VB 开发人员发出了挑战, 要求他们学习全新的概念和技术。

首先, 需要学习的是 VB.NET 和旧版本之间的区别。在一些情况下, 相同的功能要采用不同的方式来实现。这些改变不是任意的, 而是有很充足的理由。但我们必须抛弃一些旧习惯, 培养新习惯。

接着, 必须接受新概念。完全的面向对象、新的组件技术、用于本地和 Internet 界面的新可视化工具, 所有这些都是在 VB.NET 中高效开发应用程序的必备技能。

本书将全面阐述 VB.NET 2003: 首先讨论了 .NET Framework; 在本书的结尾, 讨论了部署 .NET 应用程序的最佳方式。中间部分讨论了从数据库访问到与诸如 XML 等其他技术集成的一切内容, 并且详细地介绍了新增功能。您将会看到, VB.NET 已经成为一种强大且易于使用的语言, 使您在编写网络程序时, 就如同编写桌面程序一样得心应手。

VB 的重要性

在 .NET 的早期应用中, Microsoft 的新语言 C# 得到了广泛的关注。但随着 .NET 应用的增多, VB 的重要性也逐渐显现出来。Microsoft 对外宣称, 当生产率是最重要的因素时, VB.NET 是开发应用程序的最佳语言。

VB 的未来开发将强调以最高效的方式访问整个 .NET Framework 的能力, 而 C# 开发强调的是编写代码的经验, 这非常切合 VB 的传统作用: 语言开发人员在现实世界中使用它尽可能快地创建商务应用程序。

区别不仅仅在学术方面。 .NET Framework 的一个最重要的优点是, 它允许用非常少的代码编写应用程序。在商务应用程序中, 主要目标是编写业务逻辑, 尽可能减少例程编码任务。其价值在新的 .NET Framework 中不是编写出许多代码, 而是用尽可能少的代码编写出更健壮、有效的应用程序。

VB 非常适合于这种类型的开发, 所以成为目前既经济又实惠的软件开发工具, 将来它还会为此目的做进一步的细化和改进。

本书读者对象

本书主要针对有经验的、希望深入学习 VB.NET 的开发人员。本书讨论了最常见的编程任务和概念, 既适合于从早期版本过渡到 VB.NET 版本的开发人员, 也适合于以前使用过 VB.NET, 目前需要更深入理解 VB.NET 的开发人员。

本书对 VB.NET 概念进行了全面的介绍,但 .NET Framework 的内容广博而丰富,不可能在一本书里讲清楚,本书没有把 Web 开发作为最重要的内容,但包含了 VB.NET 中基于浏览器的编程内容,专业的 Web 开发人员应该参考由清华大学出版社引进并出版的《ASP.NET 1.1 高级编程》一书。

使用本书需要的条件

为了完全而充分地利用本书,尽管可以使用包含在 .NET Framework SDK 中的命令行工具创建 VB.NET 应用程序,但仍需要 VS.NET 2003(Professional 版或更高版本),其中包含 .NET Framework SDK。可以使用 VS.NET 2002,但一些对话框和菜单会略有不同。

此外:

- 一些章节要使用 SQL Server 2000,但也可以使用 Visual Studio .NET 附带的 MSDE (Microsoft Data Engine)来运行示例代码。
- 有几章要使用 Internet Information Services (IIS),在 Windows 2003 Server、Windows 2000 Server、Windows 2000 Professional 和 Windows XP 系统中均配有 IIS,但在默认状态下是不安装的。
- 第 18 章要使用 MSMQ 来进行排队事务处理,在 Windows 2003 Server、Windows 2000 Server 和 Windows 2000 Professional 系统中均配有 MSMQ,但在默认状态下是不安装的。

本书主要内容

第 1 章解释了 .NET 的重要性和它对应用程序开发的影响。为了理解为什么需要 .NET,以及当前开发技术存在哪些缺陷,我们首先讨论了包括 COM 和 DNA 在内的体系结构模式,然后讲述 .NET 如何使用公共语言运行库(Common Language Runtime, CLR)来纠正这些技术中的缺点。

第 2 章讨论了一个 VB.NET 应用程序。在开发这个应用程序时,我们将讨论 VS.NET 的一些新功能。

第 3 章探讨了 .NET 平台的核心——公共语言运行库。CLR 负责管理为 .NET 平台编译的代码的执行。我们将论述版本变化和部署、内存管理、跨语言集成、元数据和 IL 反汇编程序。

第 4 章将讨论 Visual Basic 中的许多常用类型。本章的主要目的是让读者熟悉值类型和引用类型,帮助具有 VB 6 背景的开发人员理解 VB.NET 中定义变量的方式的主要差异。

从第 5 章开始论述 VB.NET 中面向对象编程的特性。本章将定义对象、类、实例、封装、抽象、多态性和继承性。

第 6 章讲述了继承的概念及在 VB.NET 中如何利用继承。我们将解释如何创建简单和抽象的基类,如何创建可以派生其他类的基类。

第 7 章把第 5 章和第 6 章的理论应用于实践。我们将讨论 4 个主要的面向对象概念(抽象性、封装性、多态性、继承),以及如何将这些概念应用到设计和开发中,以创建高效的面向对象的应用程序。

第 8 章介绍了命名空间及其层次结构。解释了命名空间的概念和某些常用的命名空间。此外,还介绍了如何创建新的命名空间,以及在项目中如何导入命名空间,如何给命名空间起别名。

第 0 章通过详细讨论 CLR 异常处理程序和新的 Try...Catch...Finally 结构,描述了 VB.NET 中错误处理的原理。除此之外,还描述了错误和跟踪的记录,以及如何利用这些方法获取程序工作方式的反馈。

第 10 章展示了 .NET Framework 用于 XML 的生成和操作的功 能,讲述了 .NET Framework 中与 XML 相关的命名空间,并且详细地讨论了这些命名空间所包含的类的子集。该章还概述了一系列利用 XML 的技术,特别是 ADO.NET 和 SQL Server。

第 11 章集中讨论了需要了解的 ADO.NET 对象模型,以便能够建立灵活、快速和可伸缩的数据访问对象和应用程序。该章还讲述了 ADO 向 ADO.NET 的转化,讨论了为了建立对 .NET 应用程序的数据访问而需要理解的 ADO.NET 主要对象。

第 12 章讲述了 Windows 窗体,主要讨论了窗体和内置控件。还讨论了哪些是新的,哪些是对 Visual Basic 以前版本的修改,还讨论了 System.Windows.Forms 命名空间。

第 13 章讨论了如何创建自己的 Windows 控件。特别是讨论了如何从其他控件继承,如何建立复合控件,如何根据 Control 类从头开始编写控件。

第 14 章探讨了 Web 窗体以及如何从它们的使用中受益。本章通过使用越来越复杂的示例,讲述了如何利用快速应用程序开发(Rapid Application Development,一般与 Windows 应用程序相关)的强大功能开发 Web 应用程序。

第 15 章讨论的是开发 Visual Basic 控件的一种全新形式:定制 Web 控件。它描述了 .NET Framework 中可用的各种形式的自定义 Web 控件开发,还探讨了 Web 用户的基本结构和子类控件,以及复合控件和模板控件。

第 16 章介绍了 .NET 中的数据绑定是如何把底层数据库和控件关联起来的,这比使用以前版本的 VB 容易一些。我们将论述 .NET 如何自动从底层数据库中为控件填充数据,并提供更新底层数据库的一种机制,以响应用户可能对 Windows 应用程序所做的修改。

第 17 章介绍了 COM 和 .NET 组件的互操作性,Microsoft 提供了哪些工具将这两项技术链接起来。

第 18 章探讨了 .NET 组件服务,特别讨论了事务处理和排队组件。

第 19 章将学习线程,理解 .NET Framework 中的各种对象如何让用户开发出多线程的应用程序。我们将讨论如何创建线程,如何将它们与进程关联在一起,以及多任务和多线程之间的差异。

第 20 章讨论如何在传统的 3 层应用程序设计中使用远程服务。本章将论述远程服务的基本体系结构,并建立一个基本服务器和客户程序,使用一个对象将客户的请求传递到业务层上。接着介绍如何使用串行化把服务器上更复杂的对象传递给客户,以及如何使用调用环境(call context),在每个调用中,把额外的数据从客户端传递给服务器,但无需改变对象模型。

第 21 章讨论了在 Windows 服务的生成过程中,如何使用 VB.NET。该章将讨论 Windows 服务的创建、安装、运行和调试。

第 22 章探讨了如何利用 VB.NET 创建和使用 Web 服务。该章还讨论了 CLR 提供的用于建立和利用 Web 服务的抽象类,论述了一些支持 Web 服务的技术。最后,该章讲述了使用分布式体系结构的缺点和 Web 服务的前景。

第 23 章介绍了如何从 Web 上下载资源,如何设计自己的通信协议,如何在应用程序中复用 Web 浏览器控件。

第 24 章研究为了注重安全性而由 .NET 提供的附加工具和功能。在该章中, 我们讨论了 Caspol.exe 和 Permview.exe, 以及它们如何帮助建立和维护安全策略, 还描述了 System.Security.Permissions 命名空间, 以及它与管理权限的关系。最后讨论了 System.Security.Cryptography 命名空间, 并运行一些代码说明该命名空间的功能。

第 25 章集中介绍了程序集和它们在 CLR 中的用法。在这一章里, 我们讨论了程序集的结构、程序集所包含的内容, 以及程序集所包含的信息。此外, 还讨论了程序集的清单及其在部署中的作用。我们也将学习 Visual Studio .NET 和 CLR 在部署应用程序时提供的功能。

第 26 章介绍了可移式 Web 应用程序、Mobile Web 和 .NET Compact Framework 应用程序之间的区别、.NET Compact Framework 以及可移式设备模拟器。

附录 A 介绍了 Visual Basic 兼容库, 以帮助实现现有代码的转化, 并为转移到 VB.NET 的开发人员提供向下兼容性和支持。

源代码

对于本书中的所有范例来说, 您可能更愿意亲自将所有代码都输入到计算机中。许多读者更喜欢采用这种方式, 因为这样可以更加有效地掌握那些正在使用的技术。本书中的所有源代码都放在 <http://www.wrox.com/> 网站中。在您登录到 Wrox.com 站点 <http://www.wrox.com/> 时, 只需使用 Search 工具框或使用书名列表就可以找到本书。接着单击本书信息页面上的 Download Code 链接, 就可以获得所有的源代码。

说明:

因为很多书的标题相类似, 所以最简单的方法是通过 ISBN 进行搜索。本书的英文 ISBN 号是 0-7645-5992-3。

从该站点上下载的文件是已经使用了 WinZip 进行压缩的文件。在把附件保存到硬盘的一个文件夹中时, 需要使用解压缩文件(如 WinZip 或 PKUnzip)对该文件解压缩。或者, 您可以访问 Wrox 的下载页面: [Http://www.wrox.com/dynamic/books/download.aspx](http://www.wrox.com/dynamic/books/download.aspx), 来查看本书或其他所有由 Wrox 出版的书籍中的代码。

勘误表

尽管我们已经尽了各种努力来保证文章或代码不出现错误, 但是错误总是难免的, 如果您在本书中找到了错误, 请告诉我们, 我们将非常感激。通过勘误表, 可以让其他读者避免受挫, 当然, 这还有助于提供更高质量的信息。

要在网站上找到勘误表, 可以登录 <http://www.wrox.com>, 通过 Search 工具框或书名列表查找本书, 然后在本书的信息页面上, 单击 Book Errata 链接。在这个页面上, 可以查看已经提交并通过编辑检查的所有勘误。此外, 在 www.wrox.com/misc-pages/booklist.shtml 页面上还提供了完整的书籍列表以及相关的勘误表链接。

如果在 Book Errata 页面未找到相关错误, 则可以访问 www.wrox.com/contact/techsupport.shtml 并填写其中的表单, 将您发现的错误发送给我们。我们将核查该信息, 并将该消息发布到该书的勘误表页面中, 且在后续的版本中修正该问题。

p2p.wrox.com

P2P 论坛是为作者和同行之间的讨论而设立的。我们在邮件列表、论坛和新闻组中提供“由程序员为程序员的支持”(programmer to programmer support), 还包括一对一的电子邮件支持系统。您可以相信, 您的问题不仅仅是由支持专家解答, 而且还要提供给我们邮件列表中的许多 Wrox 作者和其他业界专家。在 p2p.wrox.com 上, 您可以从许多不同的列表中获得帮助, 不仅对阅读本书有所帮助, 还可以在开发应用程序时获得帮助。

要加入该论坛, 可以遵循下面的步骤:

- (1) 进入 p2p.wrox.com 并单击 Register 链接。
- (2) 阅读使用条款并单击 Agree 按钮。
- (3) 填写表单中要求填写的信息, 以及其他其他可选信息, 然后单击 Submit 按钮, 即可加入论坛。
- (4) 您将接受到一封确认电子邮件, 其中列出如何验证您的账户和加入论坛的完整步骤。

说明:

您不加入 P2P 论坛也可浏览其中信息, 但要发布新消息, 则必须注册。

在加入该论坛后, 您即可发布新消息并回复其他人的文章。可在任意时刻通过 Web 阅读其中信息。如果需要, 也可通过电子邮件订阅论坛中特定主题的新信息, 为此, 请在论坛列表选中特定论坛名称对应的 Subscribe to this Forum 图标。

要更详细地了解如何使用 Wrox P2P 论坛, 请阅读 P2P FAQ, 其中列出了论坛软件如何工作等众多常见的问题, 单击 P2P 论坛中任意页面中的 FAQ 链接即可阅读相关内容。

目 录

第 1 章 什么是 Microsoft .NET	1
1.1 .NET 概述	1
1.1.1 未来应用广泛而深邃的平台	1
1.1.2 DNA 和 COM 的缺陷	2
1.2 .NET Framework 概述	4
1.3 公共语言运行库	5
1.3.1 主要的设计目标	5
1.3.2 元数据	7
1.3.3 多语言集成和支持	7
1.3.4 公共类型系统	7
1.3.5 命名空间	8
1.4 .NET Class Framework	8
1.5 用户和程序接口	10
1.5.1 Windows 窗体	10
1.5.2 Web 窗体	10
1.5.3 控制台应用程序	11
1.5.4 Web 服务	11
1.6 作为.NET “元语言” 的 XML	11
1.7 COM 的角色	12
1.8 永远不会改变的一些事实	12
1.9 .NET 对 VB 的影响	12
1.10 .NET 对我们的影响	13
1.10.1 编程模型谱	13
1.10.2 减少了 Internet 开发的障碍	13
1.10.3 预编写的功能库	14
1.10.4 部署更简单	14
1.11 .NET 的未来	14
1.12 小结	14
第 2 章 VB.NET 和 VS.NET 简介	15
2.1 Visual Studio .NET——入门	15
2.2 Visual Studio .NET	17
2.2.1 Solution Explorer 窗口	17

2.2.2	命名空间	18
2.2.3	新代码窗口	19
2.2.4	Properties 窗口	22
2.2.5	动态帮助	23
2.3	使用 VB.NET	23
2.3.1	在代码中设置窗体属性	23
2.3.2	AssemblyInfo.vb 文件	24
2.4	增强范例应用程序的功能	26
2.4.1	添加控件和事件处理程序	27
2.4.2	定制代码	28
2.4.3	构建配置	32
2.4.4	创建应用程序	34
2.5	VS.NET 的有用功能	37
2.5.1	任务列表	37
2.5.2	Command 窗口	38
2.5.3	Server Explorer	38
2.5.4	在 Visual Studio .NET 中记录和使用宏	39
2.6	小结	40
第 3 章	公共语言运行库	41
3.1	.NET 应用程序的元素	41
3.1.1	程序集	41
3.1.2	模块	42
3.1.3	类型	42
3.2	版本化和部署	43
3.2.1	对于版本化的更好支持	43
3.2.2	更好的部署	44
3.3	内存管理	44
3.3.1	更好的垃圾回收功能	44
3.3.2	更快的对象内存分配	49
3.3.3	GC 的优化	49
3.4	跨语言的集成	51
3.4.1	通用类型系统	51
3.4.2	元数据	53
3.4.3	对元数据的更好支持	53
3.4.4	属性	54
3.4.5	Reflection API	56
3.5	IL Disassembler	56

3.6 小结	57
第 4 章 变量和类型	58
4.1 值类型和引用类型的区别	58
4.2 值类型(结构)	60
4.3 显式转换	68
4.3.1 Option Strict、Option Explicit 和 Option Compare	68
4.3.2 执行显式转换	69
4.4 引用类型(类)	71
4.4.1 Object 类	72
4.4.2 String 类	73
4.4.3 DBNull 类	75
4.4.4 数组	76
4.4.5 集合	78
4.5 参数传递	80
4.6 装箱	81
4.7 不再使用的关键字和方法	82
4.8 小结	83
第 5 章 对象语法综述	84
5.1 面向对象的术语	85
5.1.1 对象、类和实例	85
5.1.2 对象的组成	85
5.2 使用对象	88
5.2.1 对象的声明和实例化	89
5.2.2 对象引用	91
5.2.3 取消对象的引用	91
5.2.4 前期及后期绑定	91
5.3 创建类	95
5.3.1 创建基类	95
5.3.2 构造函数方法	111
5.3.3 终止和清除	112
5.4 高级概念	115
5.4.1 重载方法	115
5.4.2 重载构造函数方法	117
5.4.3 共享方法、变量和事件	119
5.4.4 委托	123
5.4.5 类和组件	128
5.5 小结	131

第6章 继承和接口	132
6.1 继承	132
6.2 多接口	177
6.2.1 对象接口	177
6.2.2 辅助接口	178
6.3 小结	184
第7章 对象和组件的应用	185
7.1 抽象性	185
7.2 封装性	187
7.3 多态性	190
7.3.1 方法签名	190
7.3.2 实现多态性	190
7.4 继承	200
7.4.1 使用继承的场合	200
7.4.2 继承和多接口	203
7.4.3 进一步探讨	209
7.4.4 “脆弱”的基类问题	210
7.5 小结	213
第8章 命名空间	214
8.1 命名空间的定义	214
8.1.1 命名空间和引用	217
8.1.2 通用的命名空间	218
8.2 导入命名空间并指定别名	221
8.2.1 导入命名空间	221
8.2.2 为命名空间指定别名	222
8.3 创建自己的命名空间	222
8.4 小结	225
第9章 错误处理	226
9.1 VB6 中的错误处理	226
9.2 .NET 中的异常	228
9.2.1 异常的属性和方法	228
9.2.2 异常与 VB6 中 Err 对象的区别	229
9.2.3 常用的异常类型	229
9.3 VB.NET 中的结构化异常处理关键字	230
9.3.1 Try、Catch 和 Finally 关键字	230
9.3.2 Throw 关键字	231

9.3.3 抛出新的异常	232
9.3.4 Exit Try 语句	233
9.3.5 嵌套的 Try 结构	234
9.4 使用异常属性	236
9.4.1 Message 属性	237
9.4.2 InnerException 属性和 TargetSite 属性	237
9.4.3 Source 和 StackTrace 属性	239
9.4.4 GetBaseException、GetHashCode 和 Equals 方法	240
9.4.5 HelpLink 属性	241
9.5 错误记录	242
9.5.1 事件日志	243
9.5.2 写入追踪文件	246
9.6 调试与测定性能	247
9.7 小结	251
第 10 章 在 VB.NET 中使用 XML	253
10.1 XML 简介	254
10.2 XML 串行化	255
10.3 System.Xml 文档支持	261
10.4 XML 流样式的分析程序	261
10.4.1 写入 XML 流	262
10.4.2 读取 XML 流	265
10.4.3 使用 MemoryStream 对象	271
10.4.4 DOM	274
10.5 XSLT 转换	280
10.5.1 不同 XML 标准之间的 XSLT 转换	284
10.5.2 其他的 System.Xml.Xsl 类和接口	287
10.6 ADO.NET	287
10.6.1 ADO.NET 和 SQL Server 的内置 XML 特征	289
10.6.2 类型化的 DataSet 对象	291
10.6.3 生成类型化的 DataSet	291
10.7 小结	296
第 11 章 用 ADO.NET 访问数据	297
11.1 ADO.NET 体系结构	298
11.1.1 ADO.NET 组件	298
11.1.2 ADO 和 ADO.NET 之间的区别	299
11.2 .NET 数据提供程序	302
11.2.1 Connection 对象	303

11.2.2	Command 对象	303
11.2.3	DataReader 对象	304
11.2.4	DataAdapter 对象	306
11.2.5	SQL Server .NET 数据提供程序	309
11.2.6	OLE DB .NET 数据提供程序	309
11.3	DataSet 组件	309
11.3.1	DataTableCollection	310
11.3.2	DataRelationCollection	310
11.3.3	ExtendedProperties	310
11.3.4	创建和使用 DataSet 对象	311
11.3.5	ADO.NET 的 DataTable 对象	315
11.4	ADO.NET 中的连接池	316
11.5	在 ADO.NET 中使用存储过程	316
11.5.1	创建存储过程	317
11.5.2	调用存储过程	319
11.6	创建 Data Access 组件	321
11.6.1	构造函数	323
11.6.2	属性	324
11.6.3	存储过程 XML 结构	325
11.6.4	方法	325
11.7	使用 DataSet 对象绑定 DataGrid	338
11.8	小结	340
第 12 章	Windows 窗体	341
12.1	Windows 窗体的重要性	341
12.2	System.Windows.Forms 命名空间	341
12.3	将窗体用作类	343
12.3.1	通过 Sub Main 使用窗体	344
12.3.2	.NET 中的窗体定义	344
12.4	设计时的窗体	345
12.4.1	设计期间显示的网格	345
12.4.2	设置启动窗体	345
12.4.3	窗体边框	346
12.4.4	始终置顶——TopMost 属性	346
12.4.5	owned form	347
12.4.6	启动位置	348
12.4.7	窗体的透明性	348
12.4.8	可视化继承	350

12.4.9	窗体尺寸限制的设置	350
12.4.10	滚动窗体	350
12.5	运行时的窗体	351
12.6	控件	351
12.6.1	以 Tab 键切换控件的顺序	352
12.6.2	控件数组	352
12.6.3	自动调整控件的尺寸和位置	354
12.6.4	扩展的 Provider 控件	357
12.6.5	数据输入的验证	359
12.6.6	菜单	361
12.6.7	工具栏	365
12.6.8	通用对话框	367
12.6.9	拖放操作	369
12.6.10	Panel 和 GroupBox 容器控件	371
12.6.11	标准 Windows.Forms 控件的总结	372
12.6.12	已撤销的控件	376
12.6.13	使用 ActiveX 控件	377
12.6.14	其他方便的编程技巧	377
12.6.15	MDI 窗体	377
12.6.16	VB.NET 中的 MDI 示例	379
12.6.17	对话框窗体	380
12.7	小结	382
第 13 章	创建 Windows 控件	383
13.1	控件源	383
13.1.1	内置控件	383
13.1.2	现有的 ActiveX 控件	383
13.1.3	第三方控件	384
13.1.4	自定义控件	384
13.2	在 .NET 中开发自定义控件	384
13.2.1	从另一个控件继承	384
13.2.2	建立复合控件	385
13.2.3	从头编写控件	385
13.3	从现有的控件中继承	385
13.3.1	进程的概述	386
13.3.2	创建一个仅能输入数值的文本框	386
13.3.3	修改自定义控件	389
13.3.4	为自定义控件增加额外的逻辑处理	389

13.3.5	为自定义控件创建属性	390
13.3.6	其他有用的特性	393
13.3.7	为继承的控件建立自定义事件	393
13.3.8	创建一个限制所选项数量的复选列表框	394
13.4	Control 和 UserControl 基类	397
13.4.1	Control 类	397
13.4.2	UserControl 类	399
13.5	复合的 UserControl	399
13.5.1	创建一个复合的 UserControl	400
13.5.2	调整控件的大小	401
13.5.3	为控件设置最小尺寸	401
13.5.4	提供子控件的属性	401
13.5.5	完成例子的步骤	402
13.6	从头建立控件	406
13.7	在工具箱上附加一个图标	412
13.8	小结	413
第 14 章	Web 窗体	414
14.1	Web 窗体	414
14.1.1	设置环境	414
14.1.2	HelloWorld Web 窗体	414
14.2	剖析 Web 窗体	418
14.2.1	模板的表示方式	419
14.2.2	代码部分	419
14.3	一个更复杂的示例	420
14.4	ASP.NET Web 窗体的处理流程	422
14.5	Web 窗体中可用的控件	425
14.5.1	服务器端控件的概念	426
14.5.2	HTML 服务器控件	427
14.5.3	ASP.NET 服务器控件	428
14.5.4	验证控件	430
14.5.5	用户控件	432
14.6	Web 窗体中的事件	432
14.6.1	Web 窗体的生存期	432
14.6.2	事件类别	433
14.7	Web 窗体与 ASP	434
14.8	在 Web 窗体之间传送控件	436
14.9	最后一个示例	436