

微软.NET程序设计系列

Microsoft Press

- ◆ 欧美读者评价 ★★★★★
- ◆ 著名数据库专家精心编著
- ◆ 入围欧美电脑图书排行榜
- ◆ 大量可重用的范例代码
- ◆ 数据访问不再是一个问题
- ◆ 赏心悦目的版式设计
- ◆ 附开发职业规划蓝图

MICROSOFT®

ADO.NET

程序设计

[美] Rebecca M. Riordan 著

李高健 孙瑛霖 译
Visual Studio .NET产品组 审校

Microsoft
.net



清华大学出版社

微软.NET 程序设计系列

ADO.NET 程序设计

[美] Rebecca M. Riordan 著
李高健 孙瑛霖 译
Visual Studio .NET 产品组 审校

清华大学出版社

(京)新登字 158 号

内 容 简 介

.NET 框架的数据访问组件 ADO.NET 能与 XML 兼容平台上的任何一种组件协同工作。本书介绍了 ADO.NET 的对象模型以及如何使用这些模型开发数据绑定 Windows 窗体和 Web 窗体, ADO.NET 如何与 XML 交互以及如何通过 .NET 环境访问旧版本的 ADO。该书内容包括创建连接、数据命令和数据阅读器、使用数据集、数据表和数据视图、编辑和修改数据、Windows 窗体和 Web 窗体中的数据绑定、使用 XML 设计器修改数据结构、用 ADO.NET 读写 XML 以及通过 .NET 框架访问 ADO 等。本书旨在帮助初学者快速熟悉 ADO.NET 的组成和特性, 学会如何开发 Visual Basic .NET 和 Visual C# .NET 的数据库功能。该书浅显易懂, 示例丰富。

本书适合于有一定 Visual Basic 或 Visual C# 编程基础的 ADO 初学者, 对于想要学习和扩展数据库开发技术的开发者而言, 本书也是非常理想的学习工具。

Microsoft ADO.NET Step by Step

Microsoft Press

Copyright © 2002 by Microsoft Corporation

Original English Language edition published by Microsoft Press, a Division of Microsoft Corporation

All rights reserved.

No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher. For sale in the People's Republic of China only.

本书中文简体版由 Microsoft Press 授权清华大学出版社出版发行, 未经出版者书面许可, 不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号: 图字 01-2000-3535 号

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: ADO.NET 程序设计

作 者: [美] Rebecca M. Riordan

译 者: 李高健 孙瑛霖

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: 蔡 颖 李祥青

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×960 1/16 印张: 26.75 彩插页: 4 字数: 597 千字

版 次: 2002 年 7 月第 1 版 2002 年 7 月第 1 次印刷

书 号: ISBN 7-302-05663-3/TP·3340

印 数: 0001~5000

定 价: 48.00 元

Microsoft®
.net

通往.NET的成功之路由此开始……

——Visual Studio .NET为未来的十年作好了准备。
——您呢？

.NET开发人员职业技能蓝图

从下面一种或几种环境开始您的体验

使用您最擅长的语言学习Visual Studio .NET和.NET框架

技术术语:
MSF: Microsoft Solutions Framework
XSLT: extensible Style Sheet Language Transformations
DTS: Data Transformation Services
OLAP: Online Analytical Processing
MDX: Multidimensional expressions
WAP: Wireless Application Protocol
WML: Wireless Markup Language
OWC: Microsoft Office Web Components

ADO: Active Directory Interfaces
WMI: Windows Management Instrumentation
MDM: Microsoft Management
HTML: Hypertext Markup Language
E/R Diagrams: Entity/Relationship Diagrams
COO: Collaboration Data Objects
ACORD: multidimensional ADO
FORML: Formal Object Role Modeling Language

快速应用程序开发

Microsoft Visual Basic®
Microsoft Office,
Borland PowerBuilder
和Borland Delphi

Web应用程序开发

ASP, ECMAScript,
Java™ Server Pages, Perl
Server Pages, Delphi, IBM
Web sphere®, Macromedia
ColdFusion®, Microsoft
Visual InterDev

企业应用程序开发

Microsoft Foundation
Classes, Enterprise Java
Beans™, UNIX C/C++/平台,
分布式计算环境, 客户
信息和转移系统

系统开发

Microsoft Windows DNA,
服务和驱动程序, 应用程序/
嵌入系统, UNIX 后台程序
和实时系统等

其他语言和脚本开发

APL, Lisp, Script®,
COBOL, Perl, Scheme/TeL,
Eiffel及其他第三语言

Microsoft Visual Basic® .NET
流行的Visual Basic编程语言的后续阶段。它保留了Visual Basic面向对象的特点和运行方式, 并且能够使用.NET平台的全面功能

Microsoft Visual C++ .NET
一种结合了Visual Basic易用性和C++的强大性与灵活性的编程语言, 其拥有的许多功能使开发解决方案的过程变得更快、更方便, 尤其在使用COM+和Web 服务的情况下

Microsoft Visual C++ .NET
促进Windows开发, 并使开发人员创建新制的应用程序。通过C++语言, 使用.NET改造现有应用程序

其他
语言
和脚本
如COBOL, Perl

.NET框架

智能客户端

Web 移动设备 客户端 (Windows)

Web 页面设计人员
为Web应用程序设计用户界面和执行用户界面程序 (HTML, CSS, Microsoft FrontPage®, Microsoft Office和Server格式设计人员)

移动Web 界面开发人员
为使用这样组件和XML Web 服务进行通信的移动设备开发用户界面 (ASP.NET, Microsoft Mobile Information Server, Smart Device Extensions, Microsoft .NET, HTML, WML, and Wireless Markup Language)

Windows 应用程序开发人员
为可能使用远程组件和XML Web 服务进行通信的Windows 桌面应用程序开发 (Windows 桌面, COM/COM+, Visual FoxPro®, ActiveX®, 交互操作能力和Web服务)

Web 应用程序开发人员
为使用远程组件和XML Web 服务进行通信的Web 应用程序开发 (ASP.NET, Web 服务, XSLT, XSL, XML, Web Services, .NET, My Services)

移动应用程序开发人员
为可能使用远程组件和XML Web 服务进行通信的移动设备开发应用程序 (Windows CE的 .NET 压缩模型, 智能设备扩展, Web服务, Microsoft 移动设备扩展, Windows CE的 Microsoft SQL Server)

Windows 协作开发人员
为Windows 桌面开发工作流和生产力应用程序以及使用ADO.NET, Microsoft Exchange, Microsoft Office, Microsoft Office Word, Microsoft Office PowerPoint, Microsoft Office FrontPage Extensions, Microsoft Visual Studio for Applications)

XML Web Services (服务) 和服务器组件

服务器组件开发人员
开发服务组件和具有可扩展的商业逻辑性、交易和消息传递功能的XML Web 服务 (服务组件, COM+协同工作能力, IIS和ASP.NET安全, Visual Studio for Applications和 Passport)

数据组件开发人员
开发服务组件和具有可扩展的商业逻辑性、交易和消息传递功能的XML Web 服务 (ADO.NET, SQLXML, XSD映射, 数据绑定, XSLT, XSL, Microsoft SQL Server)

门户组件开发人员
开发对智能客户端具有智能服务组件提供Web服务、Web服务和XML Web 服务、Microsoft Exchange, Microsoft Office, Microsoft Office Word, Microsoft Office PowerPoint, Microsoft Office FrontPage Extensions, Microsoft Visual Studio for Applications)

为Web和移动设备中使用的数据库设计 (XSLT, Microsoft Visual Studio .NET, B2B, 集成和适配器)

为电子商务整合及数据源中使用的数据库设计 (XSLT, DTS, Host Integration Server, B2B, 集成和适配器)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

为处理公司数据开发应用程序, 通过远程过程及接受数据源数据, 以及为数据源开发应用程序 (T-SQL, ANSI SQL, DTS, 存储过程, 移动设备扩展)

数据服务

数据库设计人员

创建应用程序以便在企业范围内访问和/或提供数据 (ADO.NET, OLEDB/SQL/ODBC驱动程序, SQLXML, XSD和数据源计划映射) 访问服务

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

数据库设计人员
设计、构建和测试数据库设计, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业解决方案

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

企业级开发人员
设计、构建和测试企业级应用程序, 方案是否出于优化 (E/R Diagrams, 规范化, 索引化, 优化, First-Interprete Manager Query Optimizer)

清华大学出版社
http://www.tup.com.cn

《微软.NET 程序设计系列》序

英文版 Visual Studio .NET 是 2002 年 2 月 13 日发布的，之所以选择这一天，原因很多，但其中一个很浪漫的因素是第二天是情人节，比尔想把 Visual Studio .NET 作为礼物送给全世界的程序员。

摩尔定律告诉我们，每过 10 年计算机系统性能将会提高 100 倍，通信带宽也会增加 100 倍，而需要的资金却不会增加。在过去的 30 年中，这种情况确实发生了。由于微软和计算机界的共同努力，PC 进入了家庭，软件也变成了一个巨大的产业。软件开发在过去的数十年里发生了巨大的变化，持续不断的技术进步和创新推动软件工业发展到了一个新的阶段。

这期间，软件开发工具所起的巨大作用是有目共睹的。早期的程序开发，所使用的编程语言是机器指令，编程人员只能使用二进制代码编写程序。由于硬件条件的限制，在编写程序时考虑最多的是如何节省资源，尽可能少的使用 CPU 时间和内存。因此，程序开发难度之大是当今年轻的程序员们难以想象的。后来，技术人员发明了程序设计语言，从汇编语言到高级程序设计语言，包括结构程序设计语言，逻辑/函数程序设计语言和面向对象的程序设计语言。尤其是软件危机所引发的软件工程的革命，软件重用、组件开发等大量新技术的涌现，软件开发的效率大大提高了，同时也大大增强了软件的功能。

大家可能还记得程序设计从字符模式转换到图形界面的情形。那时硬件的速度依然很慢，人们所习惯的是 MS-DOS 的编程规则和 MS Basic、Turbo Pascal 和 Quick Basic 等编程工具。由于没有与图形界面相适应的开发工具，软件开发变得十分困难。幸运的是，Visual Basic、PowerBuilder、Delphi 等几种突破性开发工具的诞生很快就改变了这种状态，使得图形界面下的应用程序开发和字符模式下的开发一样简单。

Internet 的诞生和流行使软件设计发生了质的变化。运行在服务器上的应用程序将结果以 HTML 格式发送给浏览器，而客户端可以随时随地连接到任意一个站点，得到想要的信息。应用程序不需要同时存在于服务器和客户机。主要的应用程序都运行在服务器上，客户机只需安装浏览器。大量的软件开发工具迅速地应运而生。随着大量 Web 应用程序的开发，软件设计人员意识到，仅在一端(服务器端)有智能是不够的，人们真正需要的关键应用程序应该是两端都具有智能，即让不同的计算机针对同一问题协同工作。应该让用户看到来自许多不同 Web 站点的信息，同时容许用户对这些信息进行编辑和加以注释。

事实上，其典型应用即是电子商务。那么，如何为一个复杂的机构建立电子商务呢？比如说，如何将 A 公司连接到 B 公司？A 公司内部可能运行着许多不同的系统，有大量的员工需要与不同的公司分别从事不同的业务。而那些公司的情况也同样复杂。如何来设计这样的软件，使得有关双方可以通过数字化的方案了解彼此的进展和状态，从而与当今基于电话和纸张的沟通相比，大大地提

高工作效率呢？

这就是分布式计算。分布式计算的基本思想是将不同的系统互联互通，这在学术界已经存在很长时间了。今天，分布式计算已成为一种实实在在的需求。因此，必须有一种有效的开发工具来帮助开发人员建立这种分布式计算系统。

1996 年，微软公司和业界其他公司一起，将 SGML 结构化文档标准归纳、概括为一种新的标准——XML。这一标准使数据共享获得新的动力，但仅有 XML 还不能实现互联网上的计算机之间的“真正”相联——不能发现其他计算机所提供的服务，不能彼此发送消息，通过消息共享服务。现实的情况是，计算机之间的连接不一定可靠，双方的软件开发人员也未协调过彼此的开发，甚至另一端的计算机上可能正运行着恶意代码。即便面对这样的现实，开发人员依然必须确保他们的计算机安全可靠。这是一个巨大的挑战，一个计算机行业必须齐心协力方能面对的挑战。

迎接挑战的武器是 XML Web 服务，即建立完全基于开放标准的服务。这些标准确保在不同的系统上生成相同格式的 HTML 文件。鉴于 XML 标准已得到广泛认可，更大的竞争变成谁能以最优的方式实现这些标准。如同在图形界面世界有 Macintosh、OS2、Windows 和其他系统在竞争。众多计算机公司将竞相建立各种 Web 服务，而且这些 Web 服务是可以互操作的。为了避免 Web 服务不兼容的问题，微软公司联合 IBM 及其他在全球计算机界最有影响的 50 家公司一起组成了 WSI，即 Web 服务协调工作组织，共同制定有关 Web 服务的标准。由于这些公司在全局的代表地位和影响，从而确保了基于这些标准的 Web 服务能够很好地协调工作。

这种真正的分布式计算需要新的开发工具来支持，而它不应该只提供一种编程语言。更为重要的是，用户可以不必要重新建立他们已有的软件应用系统。

.NET 不仅仅表现为 Web 服务，其根本的目的是通过 XML Web 服务使人们可以在任何时间、任何地点、通过任何设备获取所需要的信息。通过 XML Web 服务实现计算机与手持设备、车载设备、电视等家用电器等的互联互通。

让计算机编程变得更容易，为全球软件开发人员提供功能更加强大、更加好用的开发工具，是微软公司长期坚持不懈为之努力的一项使命。Visual Studio .NET 正是基于这种蓝图而设计的，它为开发下一代以 XML Web 服务为中心的应用程序而设计，是软件开发有史以来功能最强大、最受欢迎的软件开发工具。它有许多激动人心的新功能，提供统一的集成开发环境，支持在同样的开发环境里用 Visual Basic、Visual C++、Visual C#、Visual J# 和其他数十种编程语言，编写、调试和部署各种应用程序。Visual Basic 进化为完全面向对象的编程语言；Visual C#作为一种新的编程语言集中了 C 和 C++的优点；Visual C++在保留原有十分强大的编程能力的基础上，增加了开发托管代码(Managed Code)功能。2002 年初，C# 和 CLI(Common Language Infrastructure, 通用语言架构)已被批准为 ECMA(欧洲计算机制造协会)的标准。

正是基于这些实际的需求，经微软独家授权，清华大学出版社引进并隆重推出了《微软.NET 程

程序设计系列》丛书。本丛书从最基础的知识讲起,使用浅显的语言、生动的实例演示,全面而系统地介绍了如何使用 Visual Studio .NET 和 .NET 框架包含的各种功能。旨在循序渐进地带领想了解、掌握 .NET 技术的读者进入 Visual Studio .NET 的美妙世界。一书在握,您将发现,程序开发原来是如此轻松且充满乐趣!

到目前为止,《微软.NET 程序设计系列》共包括 6 本书:

- 《Visual Basic .NET 程序设计》

介绍了 Visual Basic .NET 面向对象编程语言的特点和结构化异常处理机制,如何利用 Windows 窗体和 Web 窗体及各种控件开发用户界面,如何利用 ADO.NET 访问各种数据源,以及如何利用 ASP.NET 技术开发 Web 应用程序;在附录部分,还专门介绍了如何将用 Visual Basic 以前版本开发的应用程序升级到 Visual Basic .NET。

- 《Visual C++ .NET 程序设计》

作者在介绍 C++ 编程特点时,重点介绍了托管代码的扩展方面,如值类型、结构异常处理、委托和事件的关系,如何使用 .NET 框架提供的各种功能开发托管代码,同时也介绍了如何访问 COM 对象和 Win32 API。

- 《ASP.NET 程序设计》

重点描述了如何建立 Web 应用和 XML Web 服务。本书的作者 G. Andrew Duthie 是资深的网络开发技术方面的专家,长期从事 ASP 和 ASP.NET 有关的培训和咨询工作,具有丰富的建立 Web 应用的经验。本书首先介绍了建立 Web 应用的基础知识,以大量的实例和练习,帮助读者迅速掌握如何通过创建和应用 Web 窗体和服务器控件,建立 Web 应用和 XML Web 服务。对于有志从事开发 Web 应用和 XML Web 服务的人士,本书是一本很好的入门教材。

- 《ADO.NET 程序设计》

ADO.NET 是微软 .NET 框架中的数据访问组件,是一种新的数据访问对象模型。本书重点描述了如何访问各种数据源和建立数据集成系统。具体内容还包括:如何在 Windows 窗体和 Web 窗体中,通过 ADO.NET 绑定数据,如何与 XML 交互,以及如何访问早期的 ADO。

- 《Visual Basic .NET 和 Visual C# .NET 面向对象程序设计》

通过 Microsoft .NET 框架中的两个功能最为强大的编程语言: Visual Basic .NET 和 Visual C# .NET,全面阐述面向对象编程(OOP)思想,并借助具体示例和代码来说明面向对象编程的各种概念。

- 《Web 数据库程序设计(.NET 版)》

基础开发类书籍,介绍了如何借助 .NET 技术,构建更智能化、更高效的 Web 数据库。

ASP.NET 和 ADO.NET 都是当前最新的技术。Visual Basic 和 Visual C++ 也是全球最受欢迎的编程语言,拥有最大的用户群,《Visual Basic .NET 程序设计》和《Visual C++ .NET 程序设计》侧重介绍了这两种编程语言,以及如何用它们构建各种 Windows 桌面应用程序和 Web 应用程序。全套书的

写作遵循了循序渐进的原则，从最新技术的基本概念介绍出发，配有大量的实例和练习，逐步引导读者从一个程序设计门外汉变成程序设计高手；可读性很强，是学习掌握 Visual Studio .NET 的最佳教材。

作为 Visual Studio .NET 产品组的开发人员，我们欣喜地看到，《微软.NET 程序设计系列》将.NET 技术更快、更广地带到每个人的身边。同时也衷心希望，所有读者都能从本丛书中充分享受到.NET 的无穷魅力！

微软中国研究开发中心

Visual Studio .NET 产品组

2002 年 6 月

前 言

ADO.NET 是 Microsoft 新推出的 .NET 框架中用于数据访问的组件。Microsoft 把 ADO.NET 看作是对以往 ADO 版本的“革命性的改进”，尽管这种说法自从它发布之日起就引起了激烈的争论。不过有一点是可以肯定的，ADO.NET 的对象模型与早期的 ADO 版本确实关系不大。

事实上，无论您喜欢与否，.NET 框架都有一个看上去让人难以拒绝的地方：这是一个崭新的平台。无论是经验多么丰富的游戏玩家或程序员，学习 .NET 框架都需要投入较多的精力。从现在开始，我们都是初学者。

欢迎使用《ADO.NET 程序设计》进行学习。本书配以大量练习，介绍了 ADO.NET 的对象模型，如何使用这个对象模型开发绑定了数据的 Windows 窗体和 Web 窗体，ADO.NET 是如何与 XML 交互的，以及如何从 .NET 环境下访问旧的 ADO 版本。

本书针对您需要详细了解的 ADO.NET 的每一个问题展开了讨论。但由于篇幅所限，本书不可能穷尽所有问题。本书的目的是让大家都能够很好地理解 ADO.NET 对象——它们是怎么一回事，是如何协同运作的。本书让我们对 ADO.NET 环境有一个基本的了解，使我们能够通过进一步的学习成为这方面的专家。通过本书，我们能够清楚的是开始编译数据应用程序的时候需要做些什么。

本书指出了由于编程语言上的差异而可能出现混淆的地方，但是，本书还是希望您对 Visual Basic .NET 或者 Visual C# .NET 比较熟悉。如果对 .NET 环境还是完全陌生的，那么您应该从 Michael Halvorson 编著的《Visual Basic .NET 程序设计》(清华大学出版社，2002)或者孙永强等编著的《Visual C# .NET 程序设计基础》(清华大学出版社，2002)开始学起。

编程练习都同时提供了 Visual Basic .NET 和 Visual C# .NET 版。这两个版本的练习其实是相同的(除了语言上的一些差异)，因此只需要选择自己擅长或感兴趣的那种语言就行了。

安装和使用练习文件

为了帮助读者节省学习时间，作者提供了本书中所有操作需要用到的练习文件。在这些练习文件的帮助下，读者可以集中精力学习如何使用 ADO.NET 来进行数据访问。通过书中的指导，自己动手操作，学用并举，这显然是获取新知识最简单有效的途径，能达到事半功倍的学习效果。请访问 <http://www.wenyuan.com.cn/> 下载资源，找到本丛书的相关链接，下载打包的练习文件。



注意

适合与本书配套使用的软件是 Microsoft Visual Studio .NET Professional，请在学习之

前确保安装了适当版本的 Visual Studio。要了解版本信息, 请查看软件的产品包装, 或启动 Visual Studio, 然后在【帮助】菜单中选择【关于 Microsoft Development Environment】, 从中查找相关信息。

系统要求

为了完成本书的练习, 需要安装以下软件:

- Windows 2000 或者 Windows XP
- Microsoft Visual Studio .NET
- Microsoft SQL Server Desktop Engine (包含在 Visual Studio .NET 中)或者 Microsoft SQL Server 2000

本书和练习中的文件都用 Windows 2000 下的 Visual Studio .NET Professional 测试过。不过, 其他的 Visual Studio .NET 版本, 比如 Visual Basic .NET Standard 和 Visual C# .NET 也同样可以使用。

因为 Windows XP 家用版没有附带 Internet 信息服务(IIS), 所以用 Windows XP 家用版无法创建本地的 ASP.NET Web 应用程序(如第 12 章和第 13 章所讨论的内容)。而 Windows 2000 和 Windows XP Professional 则包括了 IIS。

安装练习文件

在您的机器上按照以下步骤安装练习文件, 以便做本书的练习时使用其中的内容:

1. 用 WinZip 或其他解压缩软件, 对下载的练习文件进行解压缩。
2. 双击StartCD.exe文件。
3. 单击【开始安装】选项。
4. 按照【开始安装】中的指导说明安装练习文件, 还需要安装Microsoft SQL Server 2000或 Microsoft SQL Server Desktop Engine (MSDE)。

使用练习文件

练习文件包含了本书中的项目和已经完成的解决方案。Finish 文件夹包含的是可以运行的解决方案。Started 文件夹包含了练习中需要的文件。

卸载练习文件

按照以下步骤可把练习文件从电脑中删除:

1. 双击 StartCD.exe 文件。
2. 单击【卸载练习文件】选项。

3. 按照【卸载练习文件】中的步骤删除练习文件。

本书的约定和特色段落

本书在版式上力求紧凑美观、便于阅读与学习。书中采用了一些统一的格式规范，以及几种特色段落，提供更简便快捷的学习指南。

约定

书中使用了一些特殊的文本格式，以帮助读者快捷地找到所需信息：

- 每个练习中都有一系列的任务。每个任务都有若干步骤。如果某个任务中只有一个步骤，那么这一步前面用实心圆点(●)表示。
- 中文屏幕项(如菜单名、对话框名、标题名及按钮名等)均用黑体方括号括起来，以与其他文字区分，如“【文件】菜单”、“【确定】按钮”等等。
- 本书使用一些特殊的字体格式来组织内容。下表列出了本书中使用的字体格式。

字体格式	表示	例子
Courier New字体	键入的程序代码	<code>` Visual Basic Public Class Book End Class // Visual C# public class Book { }</code>
正体英文	布尔值	True, true, False, false
	类名	Book, Library, Train
	控件类型	ListBox, TextBox
	数据类型	String, string, Integer, int
	事件	Click
	窗体	Form1
	命名空间	ReadBooks
	参数类型	String, string, Integer
	属性	Command Text

特色段落

书中有两类特色段落，一类带有图标，有如下几种：



提示 对可能的结果或替代方法进行说明。



要点 提供完成一项任务所需的重要信息。



注意 提供与所讨论主题相关的其他信息。

另一类是没有图标的楷体段落，这些段落提供了正文之外更深层次的信息，可能会包含调试信息、设计提示信息或者是读者可能希望进一步了解的主题。

本书每章都以“本章小结”部分结束。小结提供了对本章所介绍的语法与技巧的简要回顾。

勘误、建议和帮助

为了保证本书内容及附加练习文件的准确性，微软出版社通过多种渠道提供支持。请到微软支持网站查询勘误和附加信息，网址为：<http://mspress.microsoft.com/support/>。

如果读者对本书内容或附加练习有疑问、意见或建议，请通过下面的方法反馈到微软出版社：

电子邮件：mspinput@microsoft.com

邮政地址：

Microsoft Press

Attn: Step by Step Series Editor

One Microsoft Way

Redmond, WA 98052-6399

此外，也可向本书中文版编辑室发送电子邮件，以反馈意见。邮件地址请见本书封底上的“读者服务邮箱”。

请注意，以上地址并不提供对 Visual Studio .NET 软件的产品支持。若要获取有关产品支持的帮助信息，请访问：<http://support.microsoft.com>。

微软出版社网站

微软出版社网站的网址为：<http://www.microsoft.com/mspress>。

微软出版社在网站上提供的信息有：微软出版社书籍的完整列表、订购信息、相关事件，以及图书的附加内容等。

此外，微软公司还在网站上提供了关于 Visual Studio .NET 软件的最新信息，网址为：

<http://msdn.microsoft.com/vstudio/nextgen/>。

目 录

前言	ix
----------	----

第 I 部分 ADO.NET 入门

第 1 章 ADO.NET 入门	3
1.1 关于对象的说明	3
1.2 ADO.NET 对象模型	4
1.2.1 数据提供程序	4
1.2.2 数据集	5
1.3 把数据绑定到简单的 Windows 窗体上	6
1.3.1 把连接和数据适配器添 加到窗体中	6
1.3.2 创建数据集	10
1.3.3 把控件简单绑定到 数据集上	12
1.3.4 把数据加载到数据集中	13
1.4 本章小结	14

第 II 部分 数据提供程序

第 2 章 创建连接	19
2.1 了解连接	19
2.2 创建连接	20
2.2.1 设计时连接	20
2.2.2 在运行时创建连接	23
2.3 使用 Connection 的属性	24
2.3.1 ConnectionString 属性	25
2.3.2 使用其他连接属性	27
2.4 Connection 方法	30

2.5 处理 Connection 事件	32
2.5.1 StateChange 事件	32
2.5.2 InfoMessage 事件	34
2.6 本章小结	34

第 3 章 数据命令和 DataReader	35
3.1 了解数据命令和 DataReader	35
3.2 创建数据命令	36
3.2.1 在设计时把数据命令 添加到窗体上	36
3.2.2 在运行时创建数据命令	37
3.3 命令属性	38
3.3.1 在设计时设置 Command 的属性	40
3.3.2 在运行时设置 Command 的属性	41
3.3.3 使用参数集合	42
3.4 Command 的方法	46
3.4.1 执行命令	47
3.4.2 DataReader	48
3.5 本章小结	54

第 4 章 数据适配器	56
4.1 了解数据适配器	56
4.2 创建数据适配器	57
4.2.1 使用【服务器资源管 理器】	57
4.2.2 使用【工具箱】	58
4.2.3 在运行时创建数据适配器	59
4.2.4 预览结果	60
4.3 数据适配器的属性	61
4.3.1 数据适配器命令	63

4.3.2 TableMappings 集合	64	7.2 创建数据表	129
4.4 数据适配器的方法	66	7.2.1 创建单独的数据表	129
4.4.1 生成数据集和绑定数据	66	7.2.2 在数据集中创建数据表	133
4.4.2 Fill 方法	68	7.3 数据表属性	135
4.4.3 Update 方法	72	7.3.1 使用数据表属性	136
4.5 处理数据适配器事件	74	7.3.2 Columns 集合	138
4.5.1 OnRowUpdating 事件	74	7.3.3 Rows 集合	141
4.5.2 OnRowUpdated 事件	79	7.3.4 Constraints 集合	146
4.6 本章小结	83	7.4 数据表的方法	151
第 5 章 ADO.NET 中的事务处理	84	7.4.1 Select 方法	152
5.1 事务	84	7.4.2 数据行的方法	154
5.2 创建事务	85	7.5 数据表的事件	157
5.2.1 创建新事务	85	7.6 本章小结	157
5.2.2 创建嵌套事务	90	第 8 章 数据视图	159
5.3 使用事务	92	8.1 了解数据视图	159
5.3.1 指派事务给命令	93	8.2 创建数据视图	160
5.3.2 提交和回滚事务	93	8.2.1 使用 Visual Studio	161
5.4 本章小结	100	8.2.2 在运行时创建数据视图	162
第 III 部分		8.3 数据视图的属性	165
操作数据		8.3.1 数据列表达式	165
第 6 章 数据集	103	8.3.2 表达式排序	168
6.1 了解数据集	103	8.3.3 行状态过滤器	170
6.2 创建数据集	104	8.4 数据视图的方法	173
6.2.1 创建类型化数据集	105	8.5 本章小结	175
6.2.2 创建非类型化数据集	107	第 IV 部分	
6.3 数据集属性	109	使用 ADO.NET 对象	
6.3.1 数据集的 Tables 集合	110	第 9 章 编辑和更新数据	179
6.3.2 数据集的 Relations 集合	118	9.1 编辑和更新数据	179
6.4 数据集方法	122	9.2 数据行的状态和版本	180
6.5 本章小结	126	9.2.1 RowState 属性	180
第 7 章 数据表	128	9.2.2 行的版本	181
7.1 了解数据表	128	9.2.3 数据行的状态和版本	181
		9.3 编辑数据集中的数据	183

9.3.1 添加数据行	183	11.1.2 使用 Parse 事件	238
9.3.2 删除数据行	185	11.2 在 Windows 控件中显示数据	240
9.3.3 改变数据行的值	187	11.2.1 简化数据输入	241
9.3.4 推迟对数据行值的修改	189	11.2.2 处理数据关系	246
9.4 更新数据源	192	11.2.3 使用数据窗格显示 分层数据	248
9.4.1 使用 DataAdapter 的 Update 方法	192	11.2.4 使用 TreeView 显示 分层数据	253
9.4.2 执行 Command 对象	195	11.2.5 查找数据	256
9.5 接受和拒绝数据集修改	200	11.3 在 Windows 窗体中验证数据	263
9.5.1 使用 AcceptChanges	200	11.3.1 数据变化事件	263
9.5.2 使用 RejectChanges	204	11.3.2 控制验证事件	267
9.6 本章小结	207	11.3.3 使用 ErrorProvider 组件	271
第 10 章 Windows 窗体中的 ADO.NET 数据绑定	208	11.4 本章小结	275
10.1 了解 Windows 窗体中的数 据绑定	208	第 12 章 Web 窗体中的数据绑定	277
10.2 把控件绑定到 ADO.NET 数据源上	210	12.1 了解 Web 窗体中的数据绑定	277
10.2.1 简单绑定控件属性	210	12.2 把控件绑定到 ADO.NET 数据源	279
10.2.2 复杂绑定控件属性	216	12.2.1 简单绑定控件属性	279
10.3 使用 BindingContext 对象	219	12.2.2 复杂绑定控件属性	283
10.4 使用 CurrencyManager 对象	220	12.2.3 使用 DataBinder 对象	286
10.4.1 CurrencyManager 的 属性	220	12.3 维护 ADO.NET 对象的状态	288
10.4.2 CurrencyManager 的 方法	225	12.3.1 在服务器上维护 ADO.NET 对象	288
10.4.3 CurrencyManager 的 事件	226	12.3.2 在页面上维护 ADO.NET 对象	290
10.4.4 使用 Binding 对象	228	12.4 在 Web 窗体中更新数据源	292
10.5 本章小结	231	12.5 本章小结	295
第 11 章 在 Windows 窗体中 使用 ADO.NET	232	第 13 章 在 Web 窗体中使用 ADO.NET	296
11.1 格式化数据	232	13.1 使用基于模板的 Web 控件	296
11.1.1 使用 Format 事件	233	13.1.1 使用数据网格控件	297
		13.1.2 使用数据列表控件	309

13.2 移动数据	315
13.3 本章小结	327

第 V 部分

ADO.NET 与 XML

第 14 章 使用 XML 设计器

14.1 了解 XML 架构	331
14.2 创建 XML 架构和类型化 数据集	332
14.2.1 创建架构	333
14.2.2 创建数据集	334
14.2.3 了解架构属性	336
14.3 在 XML 设计器中使用数据表	338
14.3.1 把数据表添加到 XML 设计器中	339
14.3.2 创建键	342
14.3.3 创建关系	345
14.4 使用元素	348
14.5 使用类型	351
14.5.1 简单类型	351
14.5.2 复杂类型	353
14.6 使用特性	356
14.6.1 特性的属性	356
14.6.2 创建特性	357
14.7 本章小结	358

第 15 章 读写 XML

15.1 了解 ADO.NET 和 XML	360
15.2 使用数据集的 XML 方法	361
15.2.1 GetXml 和 GetXmlSchema 方法	361
15.2.2 ReadXmlSchema 方法	367
15.2.3 InferXmlSchema 方法	370
15.2.4 ReadXml 方法	372
15.2.5 WriteXmlSchema 方法	376
15.2.6 WriteXml 方法	378
15.3 控制写 XML 的方式	381
15.3.1 使用数据关系的嵌套 属性	382
15.3.2 使用数据列的 ColumnMapping 属性	384
15.4 XmlDataDocument 对象	387
15.5 本章小结	390

第 16 章 在 .NET 框架中使用 ADO

16.1 了解 COM 互操作性	392
16.2 在 .NET 框架中使用 ADO	394
16.2.1 建立一个对 ADO 的 引用	396
16.2.2 创建 ADO 对象	398
16.3 在 .NET 框架中使用 ADOX	405
16.4 本章小结	411