



Beginning ASP.NET 2.0

ASP.NET 2.0 入门经典 (第4版)

(美) Chris Hart 等著
John Kauffman
张楚雄 高猛 译



清华大学出版社

ASP.NET 2.0 入门经典

(第 4 版)

(美) Chris Hart 等著
John Kauffman
张楚雄 高猛 译

清华大学出版社

北 京

Chris Hart, John Kauffman et al

Beginning ASP.NET 2.0

EISBN: 0-7645-8850-8

Copyright©2006 by Wiley Publishing, Inc.

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

本书中文简体字版由 Wiley Publishing, 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。

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

ASP.NET 2.0 入门经典(第4版)/(美)哈特(Hart, C.)等著; 张楚雄, 高猛译. —北京: 清华大学出版社, 2006.8

书名原文: Beginning ASP.NET 2.0

ISBN 7-302-13324-7

I.A… II.①哈… ②张… ③高… III.主页制作—程序设计 IV.TP393.092

中国版本图书馆 CIP 数据核字(2006)第 073235 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 于 平

封面设计: 康 博

版式设计: 康 博

印 装 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: 42 字数: 1022 千字

版 次: 2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷

书 号: ISBN 7-302-13324-7/TP·8302

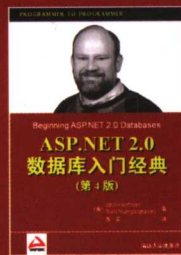
印 数: 1~5000

定 价: 78.00 元

译者简介

张楚雄，男，计算机专业硕士，曾经使用 ASP.NET 从事大型 ERP 软件的开发，目前就职于国内一家通信设备制造公司，从事系统分析工作。参与翻译了 3 本著作。

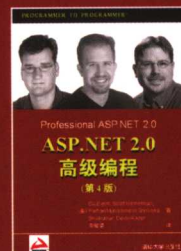
高猛，男，计算机专业硕士，高级网络系统管理员。一直从事计算机网络应用和高级程序设计方面的开发工作。参与翻译了 4 本著作。



《ASP.NET 2.0 数据库入门经典(第4版)》

John Kauffman 著
(美) Thiru Thangarathinam 译
肖奕

ASP.NET 2.0 将为 ASP 数据库领域带来深远的影响。本书全面介绍了如何使用 ASP.NET 2.0 向 Web 站点添加数据,如何将 Web 页连接至数据库。书中讲解了一系列具体示例并演示了很多现实场景和实用技巧,可有效地提升读者的 Web 数据库开发能力。



《ASP.NET 2.0 高级编程(第4版)》

Bill Evjen 等著
(美) Scott Hanselman 译
李敏波

本书在全面介绍 ASP.NET 各种编程技术的同时重点介绍了 ASP.NET 2.0 版本中的大变化,详细阐述了 ASP.NET 2.0 中所包的每个新特性。书中提供了大量的实例,帮助读者快速掌握如何在 .NET Framework 构建功能强大的 ASP.NET 应用程序。本书用于 ASP.NET 初中级程序员以及准备迁移 ASP.NET 2.0 的编程老手。



《C#入门经典(第3版)》

Karli Watson 等著
(美) Christian Nagel 译
齐立波

这是一本曾成就无数 C# 程序员的名著,厚而不“重”,可帮助您轻松掌握 C# 2005 的各种编程知识,为您的职业生涯打下坚实的基础!本书自第 1 版出版以来,全销量已达数万册,在中国也有近 4 万册的销量,已经成为广大 C# 初级程序员首选的入门教程,也是目前国内市场上最畅销的 C# 图书之一。

前言

如今，在开发复杂的 Web 站点或 Web 应用程序时，有多种技术可供选择，其中有一项技术非常出众而且使用起来非常轻松。ASP.NET 2.0 是一项非常美妙的技术，它使得开发人员能够在麻烦极少的情况下开发 Web 站点和应用程序。Web 应用程序的开发从来没有变得如此容易，虽然这项技术看起来很简单，但它的功能却非常强大，使用它可以开发出当今世界最复杂的应用程序。

Microsoft 发布的每一个新版本都提出了一种新的思考方法，新技术使得您的生活变得更加轻松，而且能够体验前所未有的编程经验；ASP.NET 2.0 也毫无例外地遵循这条规则。无论以前是否开发过 Web 应用程序，这种特别版本的技术都有很多开发人员需要学习的内容。

ASP.NET 2.0 构建在 .NET Framework 2.0 之上，它扩展了 ASP.NET 1.0 和 ASP.NET 1.1 的功能。其内核是一个基于控件的、事件驱动的架构，这意味着只需要向页面添加少量的代码，就可以看到以最小代价获得的自动结果，而且可以根据用户的输入提供平滑、直观的用户体验。

与以前的 ASP.NET 版本比较，ASP.NET 2.0 最大的变化是用户必须编写的代码量——ASP.NET 开发小组的目标是把用户要编写的代码减少 70%。根据作者长期使用 ASP.NET 2.0 的实际经验，这个数字虽然多少有些夸大，但用手指敲击键盘的数量确实减少了很多。以前必须完成的单调、重复的工作现在变得更简单。例如，为一个 Web 站点添加登录功能现在已是一个非常快捷的过程——向页面添加几个控件和建立几个用户账户就是为 Web 站点添加基本的登录功能所需进行的工作，而定制用户体验也只需要在此基础上完成少量的工作即可！

ASP.NET 2.0 中引入了一个构建应用程序的新开发环境，即 Visual Web Developer(可单独安装，也可作为 Visual Studio 2005 的一部分安装)，您将发现构建自动的、功能丰富的应用程序是一个快速且容易的过程。

Visual Web Developer 是 Microsoft 公司的一项革新，主要是根据开发人员的需求而开发的。以前的 Visual Studio .NET 版本在 Web 编程方面不是非常出色，用户经常会发现自己的代码在后台被“修改”了，因为用户的 HTML 代码和原来输入的时候看起来不一样！Visual Web Developer 带有非常强大的 HTML 编辑环境，并为开发复杂的 ASP.NET 应用程序提供了一个平滑、直观的界面。更好的是，它的价格很便宜，这使得一些不能负担整个 Visual Studio 开发包的开发人员也能购买它。

本书读者对象

本书全面介绍如何在 ASP.NET 2.0 中编写 Web 应用程序, 这些应用程序能显示存储在数据库中的数据、向用户提供定制的用户体验, 甚至可以提供购物功能。

所有这些类型的 Web 应用程序都可以使用 ASP.NET 2.0 来开发, 因此, 如果读者正打算开发这类应用程序, 那么本书将是一个很好的起点!

本书是针对 Web 编程新手, 或已知道一些 Web 编程概念的开发人员编写的。您是不是希望成为一名 Web 开发人员? 或者只是想在业余时间里学习怎样使用一些很酷的服务器端技术以便整合某些 Web 站点? 无论是哪种情况, 本书都将告诉您所需要知道的内容, 并让您充分体验到这项技术是如何工作的、怎样使用 Visual Web Developer 环境加速开发过程, 以及怎样完全控制开发过程。

本书前面的章节将帮助您快速进入 ASP.NET 开发的世界, 如果您已经具备了一定的编程基础, 那么会发现前几章阅读起来很轻松。但是, 必须注意, ASP.NET 2.0 最大限度地提供了很多技巧和工具, 我们将在本书中介绍这些内容。和 Wrox 的其他入门系列书籍一样, 您将发现在某章中介绍的概念会在其他章节中使用和扩展。

本书主要内容

本书将介绍 ASP.NET 2.0, 使用的开发工具是 Visual Web Developer IDE(Integrated Development Environment, 集成开发环境)。在阅读本书的过程中, 读者将学习如何开发强大的、数据驱动的 Web 应用程序, 以及如何使用 Web 服务发布功能。下面是本书 16 个章节的主要内容。

第 1 章: ASP.NET 2.0 和 Wrox United 应用程序简介

本章扼要介绍 ASP.NET 2.0 和 Visual Web Developer 开发环境, 并让读者创建和运行一个简单的页面。读者还将了解到 Wrox United 示例 Web 站点, 本书将使用这个示例演示 ASP.NET 2.0 的各个方面的内容。

第 2 章: 站点设计

在熟悉创建简单的页面之后, 本章讨论 Web 站点设计的概念, 并介绍 Master 页面的概念, 它使得站点中的所有页面拥有一致的外观。本章还介绍了 Web.config 和 Global.asax, 这是两个重要的 ASP.NET 文件, 它们控制着 Web 站点的行为, 以及 Web 站点地图(site map)的概念, 这个概念用于定义 Web 站点中页面的层级。

第 3 章: 页面设计

本章首先简要介绍 HTML 和 XHTML 的开发, 并介绍服务器控件这个关键概念。然后

通过在一个 Web 站点内提供导航功能来演示几个内置的服务器控件。

第 4 章：成员关系和身份验证

ASP.NET 2.0 的一个很突出的新特性是增加了 Login 服务器控件，因此本章将介绍这些控件，同时讨论如何创建用户账户、如何配置角色，以及如何激活 Web 站点的登录功能。

第 5 章：用主题设置样式

在 Web 站点的功能开发完成之后，很重要的一点是要让 Web 站点有一个合适的外观。本章介绍 CSS 样式表，并将其融入到 ASP.NET 的 Theme 功能中一起讨论，这有助于使得 Web 站点的功能和样式完全分离——对以后的维护非常有利！

第 6 章：事件和代码

对事件作出反应包括编写代码，因此本章讨论服务器端编码的概念和 Web 服务器的工作原理。在本章中，读者将了解到 HTTP 的基础知识，这样就能够理解回送(postback)，以及如何编写代码处理回送事件(postback event)。

第 7 章：读取数据

开发 Web 站点基本上都会涉及到读取存储在数据库中的数据、将数据显示在屏幕上，因此本章讨论如何使用 ASP.NET 控件(例如 GridView、DataLists 和 DetailsView 控件)连接到数据库并显示数据。另外本章还讨论从 XML 文件中读取数据。

第 8 章：写入数据

保存和更新数据是本书的下一个主题，在本章中，读者将学习几种有用的技术，这些技术用于通过参数安全地更新存储在数据库中的数据，以及通过惟一标识表项的键来引用数据。

第 9 章：代码

本章讲解基本的编程概念，如果读者希望成为一名全面的.NET 开发人员，那么必须理解这些概念。本章首先介绍基本的变量和数据类型，然后介绍集合、语句、运算符、分支和循环。最后介绍面向对象的概念和类、属性、方法以及简单的类设计原则。

第 10 章：组件化

在前面的章节中学习了编码的基本原则之后，本章将介绍编译模块，并讨论使用单独的代码文件创建页面，以及怎样将逻辑包含在不同的类或文件中以设计应用程序。另外，

本章介绍了用户控件的概念，这种控件包含的代码可以在网页上复用。

第 11 章：角色和配置文件

在第 4 章介绍的用户账户和角色的基础上，本章讨论 Web 站点设计的概念，以及根据登录 Web 站点的不同用户改变 Web 站点外观的方法。还讨论了在代码中保存用户配置文件和获取配置文件，以及根据用户的偏好切换 Web 站点的主题。

第 12 章：Web 服务

到现在为止，读者已经充分熟悉了 ASP.NET 2.0 和编码，现在应该准备好进入 Web 服务的世界了。首先，读者将学习怎样使用第三方 Web 服务，并在网页中使用该功能。接下来，读者将有机会创建自己的 Web 服务，并了解有关代理和 WSDL 的内容。

第 13 章：电子商务

在 Web 站点中添加电子商务(E-Commerce)功能可能会比较复杂，因此本章将向读者详细介绍建立在 Wrox United 示例应用程序中的电子商务实用工具，查看怎样实现一个产品目录，并创建一个和用户的配置文件相链接的购物车系统。

第 14 章：性能

如果不使用正确的方法保证 Web 站点能够在高负荷的情况下良好地运行，读者也许会发现自己 Web 站点的运行速度会越来越慢以至无法访问，因此本章将讨论多种用于增强应用程序性能的方法。这包括诸如处理对象、使用存储过程和高效使用缓存等概念。

第 15 章：错误处理

无论开发什么样的应用程序都会出现错误，因此本章讨论一些最常用的错误处理、捕获异常以及在 Web 站点中向用户显示出错页面的方法。毕竟，用户没有必要知道您的数据库服务器崩溃了，但他们希望知道，如果 Web 站点不能正确运行，故障将能在最短的时间内排除。接下来您就可以使用 ASP.NET 中的一些非常出色的调试和跟踪功能。

第 16 章：部署、构建和结束

本书的最后一章讨论如何在一个在线服务器上部署已开发完成的 Web 应用程序。所使用的示例是 Wrox United 应用程序，可以使用 Visual Web Developer 工具或称为 XCOPY 的部署工具来发布这个 Web 站点。本章还讨论了部署的测试方法以及部署时常见的问题。本章的最后部分回顾了 Wrox United 应用程序的各个不同部分，并告诉读者每个部分是在本书的什么地方进行讨论的。最后，为读者介绍了一些深入 ASP.NET 开发的书籍！

本书组织结构

本书占用大量篇幅逐步介绍各种概念，使用已编写好的示例和详细的解释，演示了 ASP.NET 应用程序的开发过程。每一章都是以前面章节的内容为基础，因此从前向后阅读本书是最好的学习方法。四位作者作为一个团队编写本书，他们之间的合作非常紧密(编辑给予了大量的帮助)，从而为读者提供了一本全面的 ASP.NET 应用程序开发基本技术手册。

使用本书所需条件

为了最大限度地利用好本书，读者应该在自己的系统上安装以下软件：

- Windows XP Home Edition 的最小化安装版本
- Microsoft Visual Web Developer
- Microsoft SQL Server 2005 Express Edition

由于 Visual Web Developer 包括了 .NET Framework 和 ASP.NET 2.0, 因此这 3 个部分就是开发 ASP.NET 应用程序所需的所有软件。

源代码

在练习本书中的示例时，读者可以手动输入所有的代码，也可以使用随书附带的源代码。本书中的所有源代码均可在 www.wrox.com 或 www.tupwk.com.cn/downpage 站点下载。登录 Web 站点 www.wrox.com 后，只需找到本书的标题(可以使用 Serach 功能或标题列表)并单击显示本书详细内容的页面上的 Download Code 链接即可获取源代码。

由于很多书籍的名称类似，因此通过 ISBN 查找可能会更容易一些；本书的 ISBN 号是 0-7654-8850-8。

在下载源代码之后，只需使用最喜欢的解压缩工具对其进行解压缩即可。另一个途径是到 Wrox 代码下载主页面 <http://www.wrox.com/dynamic/books/download.aspx>，这里有本书及其他所有 Wrox 书籍的源代码。

勘误表

我们尽最大努力确保本书在叙述和代码中没有错误。然而，没有人是完美的，错误时有发生。如果您在本书中发现什么错误，例如拼写错误或编码错误，我们将会非常感谢您能反馈给我们。通过将错误添加到勘误表中，您可以为其他读者节约数小时的时间，也可以帮助我们提供更高质量的书籍。

请给 fwkbook@tup.tsinghua.edu.cn 发电子邮件，我们就会检查您的信息，如果是正确的，我们将在本书的后续版本中采用。

要找到本书的勘误表,可以登录 Web 站点 <http://www.wrox.com>,然后搜索本书的书名或者使用标题列表。然后,在本书的详细内容页面上,单击 **Book Errata** 链接。在这个页面上读者可以查看到所有已提交的、由 Wrox 的编辑发布的错误信息。也可以在 www.wrox.com/misc-pages/booklist.shtml 页面找到一个完整的标题列表,这个列表包含了每本书的勘误表链接。

p2p.wrox.com

如果想参与讨论,可以加入 P2P 论坛,网址是 p2p.wrox.com。这些论坛是基于 Web 站点的系统,其作用是让读者发布与 Wrox 的书籍和相关技术有关的消息,并与其他读者和技术用户联络。这些论坛提供订阅功能,当有与您感兴趣的主题相关的新消息发布的时候,论坛会通个电子邮件把这些消息发送给您。Wrox 的作家、编辑、其他行业专家以及和您一样的读者都会出现在这些论坛上。

在 <http://p2p.wrox.com>,读者将找到很多不同的论坛,这些论坛不仅能帮助您阅读本书,还可以帮助您开发自己的应用程序。要加入这些论坛,可按如下步骤操作:

- (1) 登录 p2p.wrox.com 并单击 **Register** 链接。
- (2) 阅读用途条款并单击 **Agree**。
- (3) 填写加入论坛所必须的信息和可选信息并单击 **Submit**。
- (4) 您将收到一封电子邮件,该邮件将告诉您怎样验证账户并成功加入。

没有加入 P2P 论坛也可以阅读该论坛上的信息,但是如果希望发布自己的消息,则必须加入该论坛。

加入之后,您就可以发布新消息或者回复其他用户发布的消息。您可以在任何时间阅读 Web 站点上的消息。如果希望某个论坛能将最新的消息通过电子邮件发送给您,则可以单击论坛列表中该论坛名称旁边的 **Subscribe** 图标。

要获得如何使用 P2P 论坛的更多信息,可以阅读 P2P FAQ 列表中的问题及其答复,这些问题与论坛软件的工作原理以及很多与 P2P 和 Wrox 相关的常见问题有关。要阅读 FAQ,可以单击任意 P2P 页面上的 FAQ 链接。

目 录

第 1 章 ASP.NET 2.0 和 Wrox United

应用程序简介	1
1.1 将要创建的站点	2
1.2 ASP.NET 2.0——创建动态 Web 站点的强大工具	4
1.3 Web 站点常见任务的简单解决方案	4
1.3.1 一致性和个性化	5
1.3.2 导航	5
1.3.3 登录、安全和角色	5
1.3.4 连接数据	5
1.3.5 代码	6
1.3.6 组件化	6
1.3.7 Web 服务	6
1.3.8 性能和缓存	6
1.3.9 错误和异常处理	7
1.3.10 部署	7
1.3.11 部署工具	7
1.4 ASP.NET 2.0 与其他技术相融合的地方	8
1.5 Wrox United 应用程序	9
1.6 开始设计 Wrox United 站点	10
1.7 VWD Express——开发环境	10
1.7.1 Cassini	11
1.7.2 VWD 的 Solution Explorer	11
1.7.3 使用 VWD 创建、打开和使用 Web 站点及页面	13
1.7.4 示例代码目录	14
1.7.5 运行页面	16
1.7.6 设计区	17
1.7.7 工具箱	18

1.7.8 属性窗口	20
1.7.9 错误列表窗口	21
1.7.10 VWD 的数据库浏览器	24
1.8 本章小结	24
1.9 练习	25
第 2 章 站点设计	27
2.1 总体设计目标	27
2.2 Master 和 Content 页面	28
2.2.1 创建 Master 页面	29
2.2.2 创建 Content 页面	31
2.2.3 Master 和 Content 页面的示例	32
2.2.4 在 Master 页面中使用级联样式表	33
2.3 Master 页面的其他功能	38
2.3.1 多层次的 Master 页面	39
2.3.2 Master 页面支持多个内容占位符	40
2.4 创建站点地图	41
2.5 站点设计的基本原则	44
2.6 ASP.NET 2.0 应用程序的标准文件	44
2.6.1 Web.config 包含整个站点的设置	45
2.6.2 Global.asax 包含整个站点都可以使用的代码	49
2.6.3 通过 Web 浏览器编辑站点配置	49
2.7 排查站点设计中的错误	52
2.8 本章小结	52
2.9 练习	53

第 3 章 页面设计	55	5.4 本章小结.....	160
3.1 静态页面设计.....	55	5.5 练习.....	161
3.2 HTML.....	62	第 6 章 事件和代码	163
3.3 动态内容.....	67	6.1 Web 服务器体系结构.....	163
3.4 服务器端控件.....	69	6.2 服务器端事件.....	165
3.4.1 服务器控件工具箱.....	69	6.3 为页面添加事件.....	166
3.4.2 服务器控件.....	70	6.3.1 回送结构.....	171
3.4.3 VWD 的 Source View.....	74	6.3.2 可用的事件.....	172
3.5 服务器控件的类型.....	79	6.3.3 选择应该使用的事件.....	173
3.5.1 标准控件.....	79	6.4 不只是能被用户触发的事件.....	176
3.5.2 HTML 控件.....	80	6.5 间接事件.....	180
3.5.3 导航控件.....	82	6.6 取消事件.....	182
3.6 本章小结.....	94	6.7 全局事件.....	185
3.7 练习.....	95	6.8 本章小结.....	186
第 4 章 成员关系和身份验证	97	6.9 练习.....	186
4.1 安全基础知识.....	97	第 7 章 读取数据	187
4.1.1 身份——我是谁.....	97	7.1 数据库.....	187
4.1.2 身份验证——这就是我.....	98	7.2 使用 ASP.NET 2.0 的数据	
4.1.3 授权——这是我能做的.....	98	控件.....	188
4.1.4 登录站点.....	98	7.2.1 数据源控件.....	189
4.2 ASP.NET 安全性.....	99	7.2.2 数据绑定控件.....	190
4.2.1 Login 控件.....	99	7.2.3 数据源控件和数据绑定控件	
4.2.2 个性化.....	109	相配合.....	193
4.2.3 成员关系.....	111	7.2.4 使用 VWD 配置数据控件.....	194
4.2.4 身份验证.....	116	7.3 数据源控件.....	194
4.3 Wrox United 的安全性.....	118	7.3.1 数据源控件的基本属性.....	194
4.4 本章小结.....	121	7.3.2 隐藏连接字符串.....	198
4.5 练习.....	122	7.3.3 连接字符串和提供商.....	199
第 5 章 用主题设置样式	123	7.4 数据绑定控件.....	200
5.1 为站点设置样式.....	123	7.4.1 数据绑定选项列表.....	200
5.1.1 style 属性.....	124	7.4.2 GridView 控件.....	206
5.1.2 CSS——级联样式表.....	127	7.4.3 DataList 和 Repeater 控件.....	211
5.1.3 主题.....	136	7.4.4 DetailsView 和 FormView	
5.1.4 设置站点的级样式.....	152	控件.....	218
5.2 Wrox United 中的主题.....	154	7.5 带有参数的数据源控件.....	222
5.3 样式和布局的最佳实践.....	159	7.6 多个数据控件一起工作.....	226
5.3.1 可用性.....	159	7.7 XML 数据.....	232
5.3.2 可访问性.....	159	7.8 绑定语法.....	236

7.9 本章小结	236	9.7 本章小结	318
7.10 练习	237	9.8 练习	319
第 8 章 写入数据	239	第 10 章 组件化	321
8.1 写入数据的简介	239	10.1 代码与内容的分离	322
8.1.1 写入数据的控件选择	240	10.2 代码与设计的分离	323
8.1.2 DataKeyNames	240	10.3 后台代码	323
8.2 修改已存在的记录	241	10.3.1 Page 指令	324
8.3 添加新记录	246	10.3.2 不完全类	324
8.4 删除记录	249	10.3.3 事件处理程序/函数	325
8.5 上传图片	252	10.4 创建后台代码文件	326
8.6 改善图片上传功能	255	10.4.1 ASP.NET 2.0 中的编译	328
8.7 本章小结	260	10.4.2 App_Code 文件夹	329
8.8 练习	261	10.5 数据层	330
第 9 章 代码	263	10.5.1 两层应用程序	330
9.1 变量和数据类型	263	10.5.2 三层应用程序	330
9.1.1 Common Language Runtime 类型	264	10.5.3 ASP.NET 2.0 的新特性	331
9.1.2 声明变量	265	10.5.4 Wrox United Object DataSource	337
9.1.3 赋值	266	10.6 用户控件	339
9.1.4 数据转换	267	10.6.1 用户控件结构	341
9.1.5 空(Null)值	268	10.6.2 简单的用户控件	342
9.1.6 使用字符串	269	10.6.3 Wrox United News 用户 控件	346
9.1.7 使用日期	271	10.7 合成控件	351
9.1.8 使用数组和集合	274	10.8 程序集和自定义服务器控件	351
9.1.9 决定使用数组还是 使用集合	280	10.9 本章小结	351
9.1.10 枚举	280	10.10 练习	352
9.1.11 常量	282	第 11 章 角色和配置文件	353
9.2 语句	282	11.1 角色的重要性	353
9.2.1 运算符	283	11.2 Wrox United 中角色的介绍	355
9.2.2 判断	289	11.3 配置页面级别的授权	364
9.2.3 循环	292	11.4 在代码中使用角色	372
9.3 命名空间	300	11.5 启用用户配置文件	373
9.4 使用类	301	11.6 构建配置文件	374
9.4.1 创建类	302	11.6.1 Wrox United 中的配置 文件	375
9.4.2 继承	310	11.6.2 存储首选项	384
9.5 变量的作用域和生存期	316	11.7 管理匿名购物车	390
9.6 泛型	317		

11.8	本章小结	391	13.2.4	产品项页面	438
11.9	练习	392	13.3	购物车	443
第 12 章	Web 服务	393	13.3.1	购物对象	443
12.1	查看 Web 服务	394	13.3.2	配置文件	452
12.2	Web 服务的生存期	398	13.3.3	购物车控件	454
12.2.1	调用 Web 服务	399	13.4	结账	466
12.2.2	传送 Web 服务	399	13.4.1	订单处理	466
12.2.3	返回响应	401	13.4.2	登录	467
12.2.4	使用响应	403	13.4.3	地址/递送详情	467
12.3	Web 服务的结构	403	13.4.4	信用卡处理	467
12.3.1	处理指令	403	13.4.5	计划结账	469
12.3.2	命名空间	403	13.5	安全事务	482
12.3.3	公有类	404	13.6	其他工作	483
12.3.4	Web 方法	404	13.7	本章小结	483
12.4	创建 Web 服务	406	13.8	练习	484
12.5	测试 Web 服务	409	第 14 章	性能	485
12.6	Web 服务的发现	411	14.1	简单的技术	485
12.6.1	DISCO	412	14.1.1	对象处理	486
12.6.2	UDDI	412	14.1.2	数据库连接	489
12.6.3	发现 Web 服务	413	14.1.3	存储过程	489
12.7	向应用程序添加比赛 时间服务	415	14.1.4	强类型的集合	498
12.8	综合整理	417	14.1.5	会话状态	499
12.9	远程 Web 服务——PocketPC 应用程序	422	14.1.6	视图状态	499
12.10	Web 服务的安全性	426	14.2	页面和代码	500
12.10.1	加密和基于消息的 安全性	427	14.2.1	数据绑定和回送	501
12.10.2	服务的身份验证和访问 控制	427	14.2.2	对象引用	501
12.11	本章小结	427	14.2.3	StringBuilder 与字符串拼 接的对比	502
12.12	练习	428	14.2.4	挑选正确的集合	503
第 13 章	电子商务	429	14.3	缓存	504
13.1	电子商务流水线	430	14.4	性能设计	509
13.2	产品目录	430	14.5	测试性能	510
13.2.1	目录的结构	430	14.5.1	跟踪	511
13.2.2	目录的设计	431	14.5.2	压力测试工具	513
13.2.3	目录的实现	432	14.5.3	性能监控	513
			14.6	本章小结	513
			14.7	练习	514

第 15 章 错误处理	515	第 16 章 部署、构建和结束	553
15.1 防御性编码.....	515	16.1 站点部署.....	553
15.1.1 参数检查.....	516	16.1.1 检验表.....	554
15.1.2 避免假设.....	517	16.1.2 编译和运行应用程序.....	554
15.1.3 查询参数.....	518	16.1.3 发布站点.....	556
15.1.4 验证.....	519	16.1.4 XCOPY 部署.....	559
15.2 异常处理.....	524	16.1.5 在部署站点时遇到的常见问题.....	561
15.2.1 异常简介.....	524	16.2 测试和维护.....	564
15.2.2 Exception 对象.....	525	16.2.1 前面的测试.....	564
15.2.3 如何捕获异常.....	526	16.2.2 维护.....	565
15.2.4 记录异常.....	532	16.3 接下来的工作.....	571
15.2.5 邮寄异常.....	535	16.4 参考站点.....	572
15.2.6 引发异常.....	537	16.5 本章小结.....	572
15.3 异常最佳实践.....	537	16.6 练习.....	573
15.4 全局异常处理.....	538	附录 A 练习题答案	575
15.5 自定义错误页面.....	540	附录 B 配置	601
15.6 调试和跟踪.....	542	附录 C Wrox United 数据库设计	615
15.6.1 使用 ASP.NET 跟踪.....	542	附录 D VWD Database Explorer	623
15.6.2 使用调试器.....	546	附录 E CSS 和 HTML 快速参考	639
15.7 本章小结.....	552		
15.8 练习.....	552		

第 1 章

ASP.NET 2.0 和 Wrox United

应用程序简介

20 世纪后期计算机领域发生了空前的变化。对于那些以前只在卧室打转的人或终日不见阳光的瘾君子而言, Internet 的急速发展给他们带来了前所未有的乐趣。在 20 世纪 90 年代之前, 承认自己用计算机处理工作几乎是一件令人尴尬的事情, 然而突然之间人人都希望能拥有一台计算机。每种业务都争先恐后地和 Internet 扯上关系, 而且很多家庭都希望拥有自己的 Web 站点。如果一定要为 Internet 的急速发展寻找一个技术代名词, 那么这项技术非 Web 浏览器莫属。但是, 如果没有可以查看的内容, 浏览器就变得一点实用价值都没有了。用户需要信息, 而 Web 站点就像雨后春笋一样蓬勃发展, 其主题丰富多彩, 包罗万象。

20 世纪 90 年代末期, 世界充满剧变。一些商业帝国的根基就是一个最简单的想法——搜索引擎(Google)或者在线书店(Amazon)。人人都想知道怎样给自己建一个 Web 站点。HTML(HyperText Markup Language, 超文本标记语言)可以实现人们的这个愿望, 但很快人们就发现它的功能太简单了。开发人员可以显示图片和文本, 但如果您想要的不止这些, 那么应该怎么办? 如果希望站点具有反应能力, 也就是说能从用户接收信息并自动进行更新而不需要开发人员每次都编写新的页面, 应该怎么办? 如果希望将某个数据库连接到 Internet 上, 或者希望显示一个股票目录, 或者希望向每个访问站点的用户提供个性化的设置, 或者只是希望向每个访问站点的家人或朋友提供漂亮的外观, 又该怎么办?

这方面的竞争一直在继续, 人们已经开发出好几种技术, 包括 CGI、PHP 和 Java。Microsoft 参与这项竞争的技术是 ASP, 它最引人注目的特点是比其他同类技术简单, 容易入门。但是它也有很多让人激动的功能——当用户在 Web 站点内的页面之间浏览时可以保存用户的详细信息以及 Calendar 和 Rotator 等控件, 可以把这些控件直接放置到页面中, 就像使用 HTML 标记一样。ASP 获得了巨大的成功。Microsoft 做了进一步的开发: 它创建了 .NET Framework, 而 ASP.NET 则成为 ASP 技术的“升级”版本, 使用的是其成熟的编程语言 VB.NET 和 C#。这个变化在功能方面是一个很大的进步, 但是 Microsoft 在一定程度上迷失了一个很重要的目标——简单性。Web 站点的开发突然之间需要高昂的咨询费用, 同时需要顶尖的美工人员。对于那些一开始缔造站点繁荣的人而言, 它已变得高不可攀。

ASP.NET 2.0 在这个方面向正确的方向回退了一大步。Microsoft 意识到, 人们在开发 Web 站点的时候不喜欢做的一件事就是编写代码。代码很枯燥且令人讨厌。然而, Microsoft