



今日電子

美国计算机“宝典”丛书 ASP.NET Bible

丛书
累计印数
77万册

100%

内容丰富、权威

利用ASP.NET为新一代Web应用程序提供强大功能

生成、部署和运行针对任何设备的分布式应用程序

掌握使用Visual Basic .NET和C#进行ASP.NET开发

[美] Mridula Parihar 等著 王 勇 盖江南 尚红昕 等译

ASP.NET 宝典

相关网站包含
书中的所有源代码



電子工業出版社

Publishing House of Electronics Industry
www.phei.com.cn

美国计算机“宝典”丛书

ASP.NET 宝典

ASP.NET Bible

[美] Mridula Parihar 等著

王 勇 盖江南 尚红昕 等译

153.45/08

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

ASP.NET 是 .NET 框架的一部分,它是进行网络开发的重要工具之一。在原 ASP 的基础上又增加了许多新的特性,利用 ASP.NET 创建出多功能、安全和可靠的 Web 应用程序。

该书的作者在程序设计以及 Web 设计方面有着丰富的经验。该书分为 5 个部分、30 章和 6 个附录。通过范例、图解详细介绍了 ASP.NET 的特性和使用。包括如何创建自定义的 Web 控件,如何构建 Web 服务、ASP.NET 应用程序以及如何访问和处理服务器端的数据等内容。在附录中还介绍了 Visual Basic.NET 和 C# 程序设计语言的使用,为开发人员提供了参考。

本书涉及的技术全面、叙述平实易懂,适于广大 Web 应用的开发人员和维护人员使用。

Hungry Minds Copyright ©2002 by Publishing House of Electronics Industry. Original English language edition copyright ©2002 by Hungry Minds, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This translation published by arrangement with Hungry Minds, Inc. 本书中文简体专有翻译出版版权由美国 Hungry Minds, Inc. 授予电子工业出版社及其所属今日电子杂志社。未经许可,不得以任何手段和形式复制或抄袭本书内容。该专有出版版权受法律保护,侵权必究。

著作权合同登记号 图字:01-2001-2145

图书在版编目(CIP)数据

ASP.NET 宝典 / (美)帕瑞海尔 (Parihar, M.) 著;王勇等译.-北京:电子工业出版社,2002.7

(美国计算机“宝典”丛书)

书名原文:ASP.NET Bible

ISBN 7-5053-7732-9

I. A... II. ①帕... ②王... III. ①主页制作-程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2002) 第 043539 号

责任编辑:张月萍

排版制作:今日电子公司制作部

印 刷:北京东光印刷厂

出版发行:电子工业出版社 www.phei.com.cn

北京市海淀区万寿路 173 信箱 邮编:100036

经 销:各地新华书店

开 本:787 × 1092 1/16 印张:41.75 字数:1042 千字

版 次:2002 年 7 月第 1 版 2002 年 7 月第 1 次印刷

定 价:66.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。

联系电话:(010) 88211980 68279077

出版说明

21世纪是一个崭新的世纪，是催人奋进的世纪。在新世纪的第一乐章中，我们热忱地向广大读者、IT人士推介这套全新改版的美国计算机“宝典”丛书。

丛书的出版宗旨

本着提高广大读者计算机专业技能的宗旨，我社从美国 Hungry Minds（原IDGBooks Worldwide）公司引入了这套“宝典”丛书。该丛书在世界各地51个国家被译为31种文字，拥有几百万的读者。在我国自1994年引入这套丛书以来，截至2001年中期累计销量已近60余万册。得到了广大读者的认同，成为电子工业出版社的著名品牌之一。

丛书的涉及范围

“宝典”丛书的涉及范围甚广，既包括众多的流行软件、编程语言、图形图像，也包括数据库、网络等高端技术等方面的书籍。对于某些软件，我们还进行了本地化处理，按相应的中文版软件进行了调整，进一步贴近中国读者的需求。

每一本“宝典”共同贯彻的一项宗旨就是，对相应主题的介绍都非常全面、系统。使该软件或系统能做到的，读者通过本书的学习也能做到。

丛书的创作队伍

“宝典”丛书的作者都是某个计算机专业领域的专家、教授，有些还是某软件的特约测试者。比如Deke McClelland、Alan Simpson和Ellen Finkelstein等知名畅销计算机图书作家，在相关领域都具有很高的声望。他们拥有丰富的实践经验，所介绍的内容都是在工作中得到千锤百炼，具有较高的权威性。在他们所撰写的书籍当中，会介绍一些技巧，同时也会为读者提出某些忠告，以免犯同样的错误。

在中文版“宝典”中我们也本着同样的原则，所选的译者均经过严格筛选，他们大都是来自于高等院校的教授、学者，计算机领域的高手，不但具有高深的专业知识，同时也具备英语方面的深厚底蕴。我们的编辑队伍，同样是来自于计算机专业的高素质人才。通过这种严格的层层把关，我们相信最终奉献给读者的将是一部部精品。

丛书的新特性

进入新的世纪，“宝典”将以全新的面貌呈现在广大读者面前。无论是版式、用纸还是印刷质量，相关人员都颇费一番苦心，进行了很大改善。同时我们对于丛书的选题也进行了调整，使其更适合我国的计算机发展水平。对于原书中某些不适合中国国情以及过于调侃的内容进行了删减。我们将秉承“宝典”丛书一贯的“权威、全面、精益求精”的风格，力争使每一本书成为您探索计算机领域奥秘的“宝典”。

译者序

ASP.NET是当今最先进的Web开发平台,运行在服务器端。利用它可以建立功能更加强大的Web应用,它开辟了Web应用开发的新时代。作为Microsoft.NET平台的一部分,ASP.NET提供了划时代的新程序设计框架,是创建Web应用的首选平台。

与传统ASP比较而言,ASP.NET使得Web应用的开发速度大大加快,并且为开发、部署和运行Web应用提供了最便捷并且可缩放性程度最高的方式。Web应用开发人员利用少量的代码就可以为程序提供丰富的功能,从而使得应用开发更加容易,开发人员也更喜欢使用这种新式的Web开发平台。

ASP.NET在以下方面较传统的ASP有了革命性的变化:

- ◆ ASP.NET可以选择任一种可用的.NET程序设计语言,比如C#、VB.NET、JScript等等;而传统的ASP只能采用解释性脚本语言,比如VBScript、JScript语言。
- ◆ ASP.NET编写的程序执行起来速度更快。主要原因在于使用的程序设计语言的不同,ASP.NET使用编译性的程序设计语言,不需要重新编译;而ASP因为采用的是解释性语言,故每次请求时都需要编译。
- ◆ 极大地简化了部署的过程,只需将程序复制到目标Web服务器即可。
- ◆ 提供了一些对数据库访问的数据控件,比如Repeater、DataList、DataGrid控件,使得数据显示更加灵活。
- ◆ 提供了HTML和WEB两种服务器控件,增加了方法、事件驱动能力,实现HTML代码与程序的分离,便于修改和阅读。
- ◆ 提供了各种Mobile控件,使得应用程序可用于移动Web设备,比如手机。
- ◆ 采用了强大的缓存和状态管理技术,其中提供了两种缓存功能,一种对页面缓存,一种对数据缓存,从而提高了数据访问的速度。
- ◆ 采用了Code behind技术,使得界面设计与程序设计分开,便于代码的重复使用,利于程序调试和维护,提高了程序的安全性。
- ◆ 应用范围增大,不局限于Web应用。不管操作系统和程序设计语言是否相同,利用XML Web services就可以通过Internet共享数据。

本书的作者在程序设计以及Web设计方面有着丰富的经验。本书分为5个部分、30章和6个附录。通过范例、图解详细介绍了ASP.NET的基础知识、高级特性和使用。包括如何创建自定义的Web控件和设计ASP.NET数据库,如何构建Web服务、ASP.NET应用程序以及如何访问和处理服务器端的数据等内容。在附录中还介绍了Visual Basic.NET和C#程序设计语言的使用。

本书涉及的技术全面、叙述循序渐进,是Web开发人员和维护人员难得的参考书。

参加本书编译和审校工作的人员有王勇、盖江南、尚红昕、阎文丽、孙文芳、曾朝阳、韩旭、殷昊、赵迎、傅明刚、郭光、尚天任、顾群、李巍、王强、赵时颀、周宁、郝小辉、王帅等。由于时间仓促且译者水平有限,书中难免存在不妥和错误之处,恳请读者批评指正。

目 录

前言	1
第 1 部分 ASP.NET 基础知识	5
第 1 章 了解 .NET 框架	7
1.1 .NET 框架的益处	7
1.1.1 一致的编程模型	8
1.1.2 对安全性的直接支持	10
1.1.3 得到简化的开发工作	10
1.1.4 简易的应用程序实施和维护	13
1.2 .NET 框架的要素	14
1.2.1 通用语言运行时	15
1.2.2 .NET 类库	16
1.2.3 一致的组件	16
1.3 小结	18
第 2 章 开始学习 ASP.NET	19
2.1 介绍 .NET 框架	19
2.2 介绍 ASP.NET	21
2.3 建立开发环境	21
2.4 创建 ASP.NET 应用程序	24
2.4.1 创建 Visual Basic Web 应用程序	25
2.4.2 创建一个 C# Web 应用程序	27
2.5 部署 ASP.NET Web 应用程序	28
2.6 小结	30
第 3 章 利用 Web 控件建立表单	31
3.1 介绍 ASP.NET Web Form	31
3.1.1 Web Forms 组件	32
3.1.2 Web Forms 服务器控件	32
3.2 创建 Web 表单应用程序项目	33
3.3 使用 Web 控件	35
3.3.1 标签控件	36
3.3.2 文本框控件	37
3.3.3 复选框和复选框列表控件	38

3.3.4 单选钮和单选钮列表控件	40
3.3.5 列表框控件	40
3.3.6 下拉列表控件	41
3.3.7 超链接控件	41
3.3.8 表格控件	42
3.3.9 图像控件	43
3.3.10 按钮、链接按钮和图像按钮控件	44
3.4 与事件有关的工作	44
3.4.1 往返	44
3.4.2 事件处理程序	45
3.4.3 执行事件和事件处理程序	46
3.4.4 处理发送回	47
3.4.5 使用 ViewState	47
3.5 小结	48
第4章 使用 Rich Web 控件	49
4.1 使用 AdRotator 控件	49
4.1.1 AdRotator 控件的属性	50
4.1.2 AdRotator 控件的事件	51
4.1.3 向使用 AdRotator 的客户浏览器提交广告	52
4.2 使用 Calendar 控件	54
4.2.1 Calendar 控件的属性	54
4.2.2 Calendar 控件的事件	55
4.2.3 使用 Calendar 控件向客户机浏览器提交日历	57
4.3 使用 TreeView 控件	58
4.3.1 TreeView 控件的属性	59
4.3.2 TreeView 控件的事件	59
4.3.3 提交 TreeView 控件	60
4.4 使用 TabStrip 控件和 MultiPage 控件	62
4.4.1 TabStrip 控件	62
4.4.2 MultiPage 控件	63
4.4.3 一起使用 MultiPage 和 TabStrip 控件	63
4.5 使用 Toolbar 控件	65
4.5.1 Toolbar 控件的属性	65
4.5.2 Toolbar 控件的事件	66
4.5.3 提交工具栏	66
4.6 小结	68
第5章 创建及使用自定义控件	69
5.1 介绍自定义控件	69
5.2 Web 表单控件的基本结构	70

5.2.1 用于 Web 表单控件的类	70
5.2.2 用于 Web 表单控件的接口	71
5.3 创建自定义控件	72
5.3.1 创建及使用用户控件	72
5.3.2 开发复合控件	74
5.4 添加属性和方法	79
5.5 处理及暴露事件	80
5.6 小结	80
第 6 章 验证用户输入	81
6.1 理解验证控件	81
6.2 使用 RequiredFieldValidator 控件	83
6.2.1 ControlToValidate 属性	85
6.2.2 Display 属性	85
6.3 使用 CompareValidator 控件	85
6.4 使用 RangeValidator 控件	88
6.4.1 Type 属性	88
6.4.2 MinimumValue 和 MaximumValue 属性	89
6.4.3 MinimumControl 和 MaximumControl 属性	91
6.5 使用 RegularExpressionValidator 控件	91
6.6 使用 CustomValidator 控件	94
6.7 使用 ValidationSummary 控件	97
6.8 验证事件	99
6.9 多个验证控件和代码隐藏	99
6.10 使用代码	100
6.11 小结	101
第 7 章 调试 ASP.NET 页面	103
7.1 错误处理	103
7.1.1 使用自定义错误页面	104
7.1.2 跟踪错误	105
7.2 使用调试工具	106
7.2.1 Visual Studio .NET 调试器	106
7.2.2 ASP.NET 跟踪功能	110
7.3 编写好的 ASP.NET 代码	115
7.3.1 表单设计	115
7.3.2 变量和对象	115
7.3.3 编程逻辑	116
7.3.4 编码风格	116
7.4 小结	116

第 2 部分 ASP.NET 数据库程序设计	117
第 8 章 ADO.NET 概述	119
8.1 ADO.NET 概况	119
8.1.1 互操作性	120
8.1.2 可维护性	120
8.1.3 可编程性	121
8.1.4 性能	122
8.1.5 可伸缩性	122
8.2 ADO.NET 对象模型	122
8.2.1 管理提供程序	122
8.2.2 DataSet 类	124
8.3 与 ADO 进行比较	125
8.4 使用 ADO.NET 与 OLE DB 数据源进行通信	126
8.4.1 System.Data.OleDb.OleDbConnection 类	126
8.4.2 System.Data.OleDb.OleDbCommand 类	127
8.4.3 System.Data.OleDb.OleDbDataReader 类	129
8.4.4 System.Data.OleDb.OleDbDataAdapter 类	130
8.4.5 System.Data.DataSet, System.Data.DataTable, System.Data.DataRow 和 System.Data.DataColumn 类	131
8.5 小结	132
第 9 章 理解数据绑定	133
9.1 数据绑定概述	133
9.2 数据源绑定	134
9.2.1 将数据绑定到页面属性	136
9.2.2 将数据绑定到控件属性	137
9.2.3 将数据绑定到一个 ArrayList 对象	138
9.2.4 绑定方法	140
9.2.5 将数据绑定到一个 DataView 对象	141
9.3 处理发送回的数据	142
9.4 小结	143
第 10 章 数据网格的使用	145
10.1 数据网格控件的应用范例	145
10.2 设计 ASPX 页面时的附加功能	147
10.2.1 标题和脚注	148
10.3 Columns 属性的使用	152
10.3.1 BoundColumn 类	153
10.3.2 ButtonColumn 类	154

10.3.3 EditColumn 类	157
10.3.4 HyperLinkColumn 类	162
10.4 分页网格数据	164
10.5 排序网格数据	166
10.6 小结	168
第 11 章 模板的使用	169
11.1 模板概述	169
11.1.1 Repeater 控件	170
11.1.2 DataList 控件	170
11.1.3 DataGrid 控件	171
11.1.4 Repeater, DataList 和 DataGrid 三种控件的比较	172
11.2 创建模板	173
11.2.1 组合使用模板和 Repeater 控件	174
11.2.2 组合使用模板和 DataList 控件	177
11.2.3 组合使用模板和 DataGrid 控件	180
11.3 小结	182
第 12 章 在 ASP.NET 应用程序中使用 SQL Server	183
12.1 在服务器端访问来自 SQL Server 的数据	183
12.2 T-SQL 回顾	185
12.2.1 检索 SQL 数据库中的数据	186
12.2.2 在 SQL 数据库中插入、更新和删除数据	186
12.2.3 存储过程	187
12.3 在 Web 应用中实现 T-SQL	188
12.3.1 访问数据	190
12.3.2 添加数据	192
12.3.3 修改和删除数据	194
12.3.4 使用存储过程	196
12.4 使用 ADO 扩展 (ADOX)	198
12.4.1 标准 ADOX 对象	198
12.4.2 使用 ADOX 对象	199
12.5 小结	200
第 13 章 高级数据绑定和 XML	201
13.1 XML 概述	201
13.2 与 XML 相关的规范概览	204
13.2.1 文档类型定义	204
13.2.2 XML 名字空间	205
13.2.3 XML 模式	205
13.2.4 可扩展样式表语言转换 (XSL/T) 规范	206

13.2.5 XML 文档对象模型	206
13.3 ASP.NET 对 XML 的支持	207
13.3.1 System.xml 名字空间	207
13.3.2 XML Web 服务器控件	207
13.3.3 将关系型数据转换成 XML 格式	210
13.3.4 在 XML 文件中绑定服务器端控件与数据	212
13.4 小结	216
第 3 部分 高级 ASP.NET	217
第 14 章 ASP.NET 应用程序配置	219
14.1 ASP.NET 配置概念	219
14.2 Web.config 配置文件	221
14.3 ASP.NET 的配置节	223
14.3.1 <configuration> 节	223
14.3.2 <configSections> 节	223
14.3.3 <appSettings> 节	224
14.3.4 <browserCaps> 节	224
14.3.5 <compilation> 节	225
14.3.6 <customErrors> 节	227
14.3.7 <globalization> 节	228
14.3.8 <httpHandlers> 节	229
14.3.9 <httpModules> 节	230
14.3.10 Web.config 文件中的安全设置	230
14.3.11 <processModel> 节	233
14.3.12 <sessionState> 节	234
14.3.13 <trace> 会话	236
14.3.14 <webServices> 节	237
14.4 小结	239
第 15 章 开发商务对象	241
15.1 商务对象概述	242
15.2 创建和使用商务对象	242
15.2.1 创建一个 UI 为中心的商务规则对象	242
15.2.2 创建一个数据为中心的商务规则对象	243
15.2.3 使用商务对象	245
15.2.4 使用 ASP.NET 的 Data 控件创建一个数据访问组件	249
15.3 部署商务对象	255
15.3.1 创建一个单文件的程序体	256
15.3.2 创建一个多文件的程序体	256
15.3.3 创建一个部署项目	258

15.4 处理商务对象名字空间	259
15.5 小结	260
第 16 章 创建 HTTP 处理程序	261
16.1 HTTP 运行时和 HTTP 处理程序概述	261
16.1.1 ISAPI 和 HTTP 运行时	262
16.1.2 HTTP 运行时体系结构	262
16.2 用于创建 HTTP 处理程序的接口和类	264
16.2.1 IHttpHandler 接口	264
16.2.2 IHttpHandlerFactory 接口	265
16.2.3 HttpContext 类	265
16.2.4 HttpRequest 类	266
16.2.5 HttpResponse 类	266
16.3 创建 HTTP 处理程序	266
16.3.1 创建实现 IHttpHandler 接口的类	266
16.3.2 在 Web 应用程序中使用处理程序类	267
16.4 自定义 HTTP 处理程序示例	268
16.5 小结	270
第 17 章 高速缓存技术	271
17.1 高速缓存概述	271
17.2 ASP.NET 中的高速缓存	274
17.2.1 Cache API	275
17.2.2 Cache API Performance Monitor 计数器	277
17.3 高速缓存页面输出	278
17.4 高速缓存页面数据	281
17.4.1 Session 对象	281
17.4.2 Cache 对象	282
17.5 小结	287
第 18 章 使用 ASP.NET Mobile 控件创建无线应用程序	289
18.1 移动应用程序开发概述	289
18.1.1 移动应用程序开发中的问题	289
18.1.2 无线设备和仿真程序	290
18.1.3 WAP、WML 和一叠卡片	291
18.2 创建移动应用程序	292
18.2.1 静态页面	292
18.2.2 安装 MIT	295
18.2.3 创建 MIT 页面	296
18.2.4 由 WML 转化为 MIT	296
18.3 移动列表	298

18.3.1 简单列表	298
18.3.2 修饰列表	300
18.3.3 列表数据绑定	301
18.3.4 分页	301
18.3.5 打电话	304
18.4 收集用户数据	306
18.4.1 选择列表	306
18.4.2 移动文本框	308
18.4.3 使用密码或者数字	310
18.4.4 验证输入的有效性	310
18.5 小结	311
第 19 章 ASP.NET 的安全性	313
19.1 ASP.NET 安全性概述	313
19.1.1 IIS 安全性	314
19.1.2 Web 应用程序中的验证	315
19.2 基于表单的验证	317
19.3 集成安全方法	322
19.4 基于角色的安全	324
19.5 小结	326
第 20 章 本地化 ASP.NET 应用程序	327
20.1 本地化概述	327
20.1.1 字符集	328
20.1.2 配置设置	329
20.2 设置 Culture 和 Region	329
20.2.1 System.Globalization 名字空间	330
20.2.2 实现类	333
20.3 资源文件的使用	335
20.3.1 System.Resources 名字空间	336
20.3.2 创建一个认知资源的应用程序	337
20.4 小结	344
第 21 章 部署 ASP.NET 应用程序	345
21.1 部署项目概述	345
21.1.1 选择部署项目的类型	345
21.1.2 创建部署项目	346
21.1.3 在部署项目中使用编辑器	347
21.1.4 构建部署项目	359
21.2 ASP.NET 配置系统	359
21.3 部署类和程序集	361

21.4 小结	362
第 4 部分 创建 Web 服务	363
第 22 章 Web 服务概述	365
22.1 了解 Web 服务	365
22.1.1 什么是 Web 服务	366
22.1.2 Web 服务的基本元素	367
22.2 表示数据	367
22.2.1 什么是 XML	367
22.2.2 XML 语法	368
22.2.3 XML 文档结构	368
22.2.4 XML 名字空间	369
22.2.5 XML 模式	370
22.2.6 XSD 语法	371
22.3 交换消息	373
22.3.1 采用 SOAP 协议交换消息	373
22.3.2 用 HTTP-GET 和 HTTP-POST 交换消息	374
22.4 描述 Web 服务性能	375
22.5 发布和寻找 Web 服务	378
22.6 寻找 Web 服务提供商	379
22.7 Microsoft HailStorm	381
22.8 小结	382
第 23 章 Web 服务底层结构	383
23.1 Microsoft Web 服务平台	383
23.2 Microsoft .NET 框架	384
23.2.1 CLR	384
23.2.2 .NET 框架类库	385
23.2.3 ASP.NET	386
23.3 Web 服务底层结构	389
23.3.1 Web 服务目录	389
23.3.2 Web 服务发现	390
23.3.3 Web 服务描述	391
23.3.4 Web 服务有线格式	392
23.4 在 Web 服务中利用 ASP.NET 特性	393
23.4.1 支持事务	393
23.4.2 启用会话状态	394
23.4.3 Web 服务高速缓存	395
23.4.4 缓冲服务器响应	396
23.5 ASP.NET Web 服务内部	396

23.6 小结	397
第 24 章 理解 SOAP	399
24.1 什么是 SOAP	399
24.2 SOAP 规范	400
24.3 SOAP 消息元素	401
24.3.1 SOAP 包封	401
24.3.2 SOAP 标头	402
24.3.3 SOAP 正文	403
24.4 SOAP 数据类型支持	403
24.5 SOAP 异常	405
24.6 用 HTTP 作为 SOAP 消息的传送协议	406
24.7 .NET 框架中的 SOAP	408
24.7.1 使用 SOAP 标头	408
24.7.2 使用 SOAP 扩展	411
24.7.3 处理 SOAP 异常	412
24.8 Microsoft SOAP 工具包	413
24.8.1 工具包特性	414
24.8.2 创建 Web 服务	414
24.8.3 创建 Web 服务客户端应用程序	415
24.8.4 WSDL/WSML 生成器工具	416
24.8.5 SOAP 跟踪实用程序	416
24.9 小结	416
第 25 章 创建 Web 服务	417
25.1 温度转换 Web 服务	417
25.1.1 温度转换公式	417
25.1.2 方法说明	418
25.1.3 方法参数	418
25.1.4 方法行为	418
25.2 创建 Web 服务	418
25.2.1 开始创建	419
25.2.2 实现 Web 服务	424
25.2.3 构建 Web 服务	428
25.3 测试 Web 服务	428
25.3.1 使用 HTTP-GET 协议测试 Web 服务	429
25.3.2 使用 HTTP-POST 协议测试 Web 服务	433
25.4 调试 Web 服务	434
25.4.1 允许 ASP.NET Web 服务启用调试模式	434
25.4.2 启动 ASP.NET 运行时	434
25.4.3 调试 CTemp Web 服务	435

25.4.4 设置断点	436
25.4.5 测试 Web 服务方法	436
25.4.6 检查程序变量	437
25.4.7 继续方法执行过程	438
25.5 小结	438
第 26 章 部署和发布 Web 服务	439
26.1 部署准备	439
26.1.1 Web 服务名字空间	439
26.1.2 设置 CTemp Web 服务的名字空间	440
26.2 部署 Web 服务	441
26.2.1 随 Web 服务一起部署的文件	441
26.2.2 Web 服务部署工具	442
26.3 发布 Web 服务	443
26.3.1 用 DISCO 发布	444
26.3.2 用 UDDI 发布	445
26.4 小结	450
第 27 章 查找 Web 服务	451
27.1 Web 服务发现	451
27.1.1 用 Disco 工具查找 Web 服务	452
27.1.2 用 UDDI 查找 Web 服务	454
27.2 Web 服务查询和代理类	458
27.2.1 用 WSDL 工具创建代理类	458
27.2.2 用 Visual Studio 创建代理类	463
27.3 小结	464
第 28 章 使用 Web 服务	465
28.1 Web 服务客户概述	465
28.2 Web 服务代理类	466
28.3 创建客户应用程序	466
28.3.1 获得 WSDL 文档	467
28.3.2 生成代理类	468
28.3.3 创建 Web 表单	473
28.3.4 创建代理类的实例	475
28.3.5 调用 CTemp 代理方法	476
28.4 测试客户应用程序	477
28.5 应用程序执行模型	478
28.6 小结	479

第 5 部分 创建 ASP.NET 应用程序	481
第 29 章 ASP.NET Blackjack 游戏程序	483
29.1 Blackjack 游戏的规则	483
29.2 玩游戏	484
29.3 设计 Blackjack 程序	487
29.3.1 遍历程序过程	487
29.3.2 选出对象并划分责任	488
29.3.3 用户界面	488
29.4 运作环境	488
29.5 Card 对象	489
29.6 Shoe 对象的数据结构	489
29.6.1 可随机排列的堆栈	489
29.6.2 新类和堆叠填充纸牌的地方	490
29.6.3 进栈、出栈	491
29.6.4 没有取数功能	491
29.6.5 使次序具备随机性	491
29.6.6 保存和编译这个新类	492
29.7 Shoe 对象	492
29.7.1 为 Shoe 对象类编写代码	492
29.7.2 随机堆叠的地方	495
29.7.3 新的构造函数	495
29.7.4 计数、洗牌和抽牌	497
29.7.5 保存和编译 Shoe 类	497
29.7.6 Shoe 类小结	497
29.8 Blackjack 页面	498
29.8.1 游戏起始代码	502
29.8.2 Participant (参与者) 类	503
29.8.3 页面正文	505
29.8.4 初始化和对象的巧妙处理	507
29.8.5 响应事件	509
29.9 需要改进的地方	512
29.10 小结	512
第 30 章 Chatty 论坛	513
30.1 快速浏览 Chatty 论坛	513
30.2 设计 Chatty 论坛应用程序	516
30.2.1 层次结构还是无层次结构	516
30.2.2 存储讨论内容	517
30.3 创建数据库	517
30.4 填充数据库	518