

SAMS
**Teach
Yourself**

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- 秉承Read Less, Do More (精读多练)的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计, **1**小时轻松阅读学习,
24小时彻底掌握关键知识

每章**案例与练习题**助你轻松完成常见任务,
通过**实践**提高应用技能, 巩固所学知识

ASP.NET 4

入门

[美] Scott Mitchell 著
陈武 袁国忠 译



人民邮电出版社
POSTS & TELECOM PRESS

ASP.NET 4

入门经典

[美] Scott Mitchell 著
陈武 袁国忠 译



人民邮电出版社
北京

内容提要

本书通过简明的语言、详细的步骤和大量的示例，以循序渐进的方式帮助读者迅速掌握使用 ASP.NET 开发网站所需的基本知识。无论读者是否有 HTML 和编程语言方面的经验，阅读本书后都将具备使用 ASP.NET 创建数据驱动的动态 Web 应用所需的知识。

全书分四部分共 24 章。第一部分介绍了 ASP.NET 及其编程模型、Visual Web Developer 2008 速成版、ASP.NET 网页的设计、创建和测试、Visual Basic 基本知识（变量、运算符、控制结构、对象等）以及用于显示文本的 ASP.NET Web 控件；第二部分阐述用户输入的收集和处理，介绍了文本框、下拉列表、单选按钮、复选框等收集用户输入的控件以及验证用户输入的控件；第三部分讨论数据库，介绍了用于访问数据的数据源 Web 控件和用于显示数据的数据 Web 控件，讨论了数据的删除、插入和编辑以及数据绑定和其他相关主题。第四部分介绍网站导航、用户管理、母版页、使用 Ajax 提供网页的响应速度以及网站部署等。

本书适合没有 ASP.NET 经验的新手阅读，也可供使用过 ASP.NET 的程序员快速掌握 ASP.NET 4 时参考。

前言

我首次接触万维网是在 20 世纪 90 年代后期的大学时代，那时即使开发最基本的网站，也只有专业的计算机程序员能够胜任。而现在凭借 Microsoft ASP.NET 等技术，即使创建数据驱动的功能强大的动态 Web 应用程序，也是再简单不过的了。在接下来的 24 章中，您将发现使用 ASP.NET 创建真实的网站既有趣又简单。

ASP.NET Web 应用程序由不同的 ASP.NET 网页组成。正如读者将在本书的无数示例中见到的，这些 ASP.NET 网页可以显示 HTML、收集用户输入以及与数据库交互。ASP.NET 网页包含 HTML 标记和源代码，其中的源代码支持更高级的功能，如访问数据库中的数据或发送电子邮件。ASP.NET 网页的源代码可用多种编程语言编写，本书将使用 Microsoft 公司的 Visual Basic 编程语言。即使从来没有使用过 Visual Basic 甚至从来没有编程的读者也不用担心。从第 5 章开始，将用三章的篇幅介绍编程语言概念和 Visual Basic 句法。

Visual Web Developer 是一款用于创建和测试 ASP.NET 网页的免费编辑器，可在 Microsoft 官方网站下载。它简化了创建 ASP.NET 网页的 HTML 标记和源代码的过程。ASP.NET 网页的 HTML 标记可使用所见即所得 (WYSIWYG) 的图形编辑器快速创建，只需单击几次鼠标，就可将各种 HTML 元素拖放到 ASP.NET 网页并移动它们。另外，Visual Web Developer 还提供了有助于创建 ASP.NET 网页源代码的工具。

针对的读者及涵盖的内容

本书是针对 ASP.NET 新手编写的，不管是否有使用 HTML 或编程语言的经验，阅读本书后，读者都将能够使用 ASP.NET 创建并部署数据驱动的动态 Web 应用程序。

本书分四部分，总共包括 24 章。第一部分简要地介绍了 ASP.NET、HTML、Visual Web Developer 和 Visual Basic。第 1 章首先概述了 ASP.NET，然后引领读者安装 .NET 框架、Visual Web Developer 以及其他必需的组件；第 3 章简要地介绍了 Visual Web Developer，这是一个功能强大的编辑器，本书将使用它创建 ASP.NET 网页；第 5~7 章探讨了编程语言 Visual Basic 的语法和语义。

ASP.NET 提供了各种用于收集用户输入的用户界面元素，这包括文本框、复选框、下拉列表和单选按钮。第二部分将介绍如何收集和處理用戶輸入。第 10 章介绍了如何使用单行文本框、多行文本框和密码文本框；第 11 章讨论了其他用户输入控件。

第三部分演示了使用 ASP.NET 创建数据驱动的网站非常容易。从第 13 章开始，将介绍如何创建与数据库交互的网站。数据驱动的网站让用户能够在 ASP.NET 网页中查看、更新和删除数据库中的数据以及在数据库中插入数据。第 15 章将介绍如何在网页中显示数据库数据；第 16 章将探讨如何编辑、插入和删除数据。

第四部分阐述了 ASP.NET 和 Visual Web Developer 提供的可帮助创建易于使用的专业网站的工具。第 20 章将介绍如何定义网站的导航结构以及如何提供菜单、树视图和面包条；第 21 章将介绍母版页，它让 Web 设计人员能够创建网页模板并将其用于网站的所有网页。

本书使用的约定

本书使用一些设计元素和约定来帮助读者区别和参考其中包含的信息。

By the Way

注意：提供了有用的信息，读者可立即或以后阅读，而不会脱离当前的主题。

Did you Know?

提示：提供了有助于提高 Visual Basic 编程效率的信息。

Watch Out!

警告：提醒读者在特定情况下可能发生的问题或副作用。

另外，本书还使用不同的字体来帮助区分代码和正文。代码使用等宽字体表示；占位符用于表示需要读者输入的单词或字符，用斜体等宽字体表示；提示读者输入文本时，这些文本将用粗体等宽字体表示。

书中有些代码语句过长，一行容纳不下。在这种情况下，将使用续行字符（`>`）表示下一行是当前语句的继续。另外，程序清单中包含行号，这旨在方便引用特定的代码行，它们并非代码的组成部分。

衷心希望读者阅读本书时能享受到像我编写本书时的乐趣。

编程愉快！

Scott Mitchell
mitchell@4guysfromrolla.com

目 录

第一部分 ASP.NET 4 概述

第 1 章 ASP.NET 4 入门..... 2

- 1.1 什么是 ASP.NET3
 - 1.1.1 提供静态网页4
 - 1.1.2 提供动态网页4
 - 1.1.3 托管 ASP.NET 网页5
- 1.2 安装 ASP.NET Framework、
Visual Web Developer 和
SQL Server 20086
- 1.3 Visual Web Developer 简介8
 - 1.3.1 新建 ASP.NET 网站9
 - 1.3.2 创建一个简单的 ASP.NET
网页10
 - 1.3.3 测试 ASP.NET 网页12
- 1.4 总结12
- 1.5 问与答13
- 1.6 作业13
 - 1.6.1 测验13
 - 1.6.2 答案14
 - 1.6.3 练习14

第 2 章 理解 ASP.NET 编程模型 15

- 2.1 ASP.NET 网页的 HTML
部分15

- 2.1.1 HTML 简要入门 16
- 2.1.2 使用 Visual Web Developer
创建 ASP.NET 网页的
HTML 部分 18

- 2.2 探讨 ASP.NET 网页的源代码
部分28
 - 2.2.1 面向对象编程快速入门29
 - 2.2.2 ASP.NET 中的事件处理
程序31
 - 2.2.3 以编程方式使用 Web
控件32
- 2.3 总结34
- 2.4 问与答34
- 2.5 作业35
 - 2.5.1 测验35
 - 2.5.2 答案35
 - 2.5.3 练习35

第 3 章 使用 Visual Web Developer 37

- 3.1 新建网站37
 - 3.1.1 选择网站模板38
 - 3.1.2 指定网站的位置38
 - 3.1.3 选择源代码编程语言39
- 3.2 打开现有网站39
- 3.3 操作网页和其他内容40
 - 3.3.1 在网站中添加内容40
 - 3.3.2 移动、重命名和删除内容42

3.4	定制 Visual Web Developer 环境	43	5.2	声明和使用变量	70
3.5	查看、移动和调整窗口	45	5.2.1	给变量赋值	71
3.6	触手可及的帮助信息	45	5.2.2	声明变量	71
3.7	总结	46	5.2.3	变量命名规则	72
3.8	问与答	46	5.2.4	变量的数据类型	72
3.9	作业	47	5.2.5	Dim 语句	74
3.9.1	测验	47	5.3	Visual Basic 运算符	75
3.9.2	答案	47	5.3.1	算术运算符	75
3.9.3	练习	48	5.3.2	比较运算符	76
第 4 章	设计、创建和测试 ASP.NET 网页	49	5.3.3	拼接运算符	77
4.1	确定设计需求	49	5.3.4	Visual Basic 的赋值运算符	77
4.1.1	明确表述金融计算器的功能	50	5.4	Visual Basic 的数据类型规则	79
4.1.2	确定用户界面	50	5.5	总结	80
4.2	创建用户界面	51	5.6	问与答	80
4.2.1	添加三个 TextBox Web 控件	51	5.7	作业	81
4.2.2	添加 Compute Monthly Cost 按钮	53	5.7.1	测验	81
4.2.3	创建一个 Label Web 控件用于输出	54	5.7.2	答案	81
4.3	编写 ASP.NET 网页的源代码	55	5.7.3	练习	81
4.3.1	读取 TextBox Web 控件中的值	56	第 6 章	使用 Visual Basic 控制结构管理程序流程	82
4.3.2	完整的源代码	57	6.1	理解控制结构	82
4.4	测试金融计算器	58	6.2	条件控制结构	83
4.4.1	查看生成的源代码	58	6.2.1	当条件为 False 时执行指令	85
4.4.2	测试错误输入	62	6.2.2	当条件为 False 时执行另一条 If 语句	86
4.5	查看源代码	62	6.3	使用 Visual Basic 的循环控制结构	87
4.6	使用调试器	63	6.3.1	使用 For...Next 循环	88
4.7	总结	65	6.3.2	Do...Loop 循环	89
4.8	问与答	66	6.4	模块化控制结构: 子程序和函数	89
4.9	作业	66	6.4.1	使用子程序和函数减少代码冗余	92
4.9.1	测验	66	6.4.2	传递参数给子程序或函数	93
4.9.2	答案	67	6.4.3	从函数返回值	95
4.9.3	练习	67	6.4.4	事件处理程序属于哪种情况	97
第 5 章	Visual Basic 变量和运算符	69	6.5	总结	97
5.1	编程语言的用途	70	6.6	问与答	98

6.7	作业	99	9.2.2	测试 ASP.NET 网页 BMICalculator.aspx	130
6.7.1	测验	99	9.2.3	Web 窗体记住用户 输入值	131
6.7.2	答案	99	9.2.4	编写 BMICalculator.aspx 的源代码部分	133
6.7.3	练习	99	9.3	总结	135
第 7 章	在 Visual Basic 中使用对象	101	9.4	问与答	135
7.1	重新审视类和对象的角色	101	9.5	作业	136
7.2	创建对象	103	9.5.1	测验	136
7.3	设置对象的属性	104	9.5.2	答案	137
7.4	调用对象的方法	104	9.5.3	练习	137
7.5	为对象的事件创建事件处理 程序	105	第 10 章	使用文本框收集输入	138
7.6	总结	106	10.1	TextBox Web 控件基础	138
7.7	问与答	106	10.2	创建多行文本框和密码 文本框	141
7.8	作业	107	10.2.1	使用多行文本框	142
7.8.1	测验	107	10.2.2	使用密码文本框	143
7.8.2	答案	107	10.3	TextBox Web 控件的属性	145
7.8.3	练习	107	10.3.1	指定文本框的宽度	145
第 8 章	用于显示文本的 ASP.NET Web 控件	108	10.3.2	限制用户可以输入到 文本框中的字符数	146
8.1	用于显示文本的 Web 控件	108	10.3.3	格式属性: 修改文本框的 字体和颜色	147
8.2	使用 Web 控件 Literal	109	10.4	总结	148
8.3	使用 Web 控件 Label	112	10.5	问与答	149
8.4	总结	118	10.6	作业	150
8.5	问与答	118	10.6.1	测验	150
8.6	作业	119	10.6.2	答案	150
8.6.1	测验	119	10.6.3	练习	150
8.6.2	答案	119	第 11 章	使用下拉列表、单选按钮和 复选框收集输入	152
8.6.3	练习	120	11.1	不同类型的用户输入	152
第 2 部分	收集并处理用户输入		11.2	DropDownList Web 控件	154
第 9 章	Web 窗体基础	122	11.2.1	在 DropDownList Web 控件 中添加选项	154
9.1	在 HTML 网页中收集用户 输入	122	11.2.2	在 ASP.NET 网页中添加 DropDownList Web 控件	155
9.1.1	HTML 元素<input>	123	11.2.3	DropDownList Web 控件的 格式属性	159
9.1.2	使用<form>元素将输入 传回 Web 服务器	124			
9.2	剖析 ASP.NET Web 窗体	127			
9.2.1	添加 Web 控件以收集用户 输入	128			

11.3	使用 RadioButton Web 控件 从一系列选项中选择一个.....160
11.3.1	使用属性 Text 和 GroupName 162
11.3.2	确定选择了哪个 RadioButton Web 控件..... 163
11.3.3	格式属性..... 164
11.4	使用 CheckBox Web 控件.....164
11.5	总结167
11.6	问与答167
11.7	作业168
11.7.1	测验 168
11.7.2	答案 168
11.7.3	练习 169
第 12 章	使用验证控件验证用户输入 .. 170
12.1	为何需要验证用户输入170
12.2	在 ASP.NET 网页中验证 用户输入172
12.3	RequiredFieldValidator 验证 控件174
12.3.1	指定验证 Web 控件要 验证哪个 Web 控件 174
12.3.2	指定输入无效时显示的 错误消息..... 176
12.3.3	测试 ASP.NET 网页 176
12.3.4	以编程方式确定用户输入 是否有效..... 178
12.3.5	基本验证控件功能总结 .. 179
12.4	CompareValidator 验证控件.....180
12.5	使用 RangeValidator.....184
12.6	使用 RegularExpression Validator 验证输入186
12.7	验证 Web 控件的格式 属性187
12.8	其他验证控件简介189
12.9	总结190
12.10	问与答190
12.11	作业191
12.11.1	测验 191
12.11.2	答案 191
12.11.3	练习 192

第三部分 使用数据库

第 13 章	数据库简介 194
13.1	数据库基础..... 194
13.2	存储结构化数据..... 196
13.2.1	表中的列 196
13.2.2	主键列 197
13.3	新建数据库..... 198
13.4	创建数据库表.....200
13.5	在 Books 表中添加数据204
13.6	总结207
13.7	问与答.....207
13.8	作业207
13.8.1	测验.....207
13.8.2	答案.....207
13.8.3	练习208
第 14 章	使用数据源 Web 控件访问 数据 209
14.1	数据源控件.....209
14.1.1	使用 SqlDataSource 控件210
14.1.2	查看 SqlDataSource 控件的 HTML 标记215
14.2	数据库语言 SQL 简介215
14.3	深入探讨 SQL SELECT 语句.....216
14.3.1	在 Visual Web Developer 中查看 SQL 查询结果.....216
14.3.2	使用 WHERE 子句限制 返回的记录.....219
14.3.3	使用 ORDER BY 子句对 结果排序.....221
14.4	在 SqlDataSource 控件向导中 对数据进行筛选和排序222
14.4.1	筛选 SqlDataSource 控件的 数据 222
14.4.2	对 SqlDataSource 控件的 数据排序224
14.4.3	测试包含 WHERE 子句的 查询224
14.4.4	查看 SqlDataSource 控件的 HTML 标记225

14.5	总结	226	16.3.2	创建可编辑的 GridView	256
14.6	问与答	226	16.3.3	定制编辑界面和 更新规则	258
14.7	作业	227	16.4	使用 DetailsView 插入数据	266
14.7.1	测验	227	16.5	总结	267
14.7.2	答案	227	16.6	问与答	268
14.7.3	练习	228	16.7	作业	268
第 15 章	使用数据 Web 控件显示 数据	229	16.7.1	测验	268
15.1	数据 Web 控件概述	229	16.7.2	答案	269
15.2	使用 GridView 控件显示 数据	232	16.7.3	练习	269
15.2.1	GridView 的声明标记	233	第 17 章	使用数据绑定 DropDownList、 RadioButton 和 CheckBox	272
15.2.2	定制 GridView 的外观	234	17.1	列表 Web 控件概述	272
15.3	使用 DetailsView 每次显示 一条记录	238	17.1.1	将数据绑定到列表 Web 控件	273
15.3.1	定制分页界面	240	17.1.2	动态填充列表控件的 优点	274
15.3.2	定制 DetailsView 的 外观	241	17.1.3	以编程方式响应选择 变化	275
15.4	使用 GridView 进行分页和 排序	241	17.2	使用 DropDownList 筛选 结果	278
15.4.1	分页	242	17.2.1	在 DropDownList 中列出 类型	279
15.4.2	提供可排序数据	243	17.2.2	根据选定类型筛选 数据	280
15.5	总结	245	17.3	使用 CheckBoxList 和 RadioButtonList 收集用户 输入	281
15.6	问与答	245	17.3.1	遍历列表 Web 控件的 列表项	282
15.7	作业	246	17.3.2	使用 SelectedItem 和 SelectedValue	283
15.7.1	测验	246	17.3.3	定制 RadioButtonList 和 CheckBoxList 控件的 外观	284
15.7.2	答案	246	17.4	总结	285
15.7.3	练习	246	17.5	问与答	285
第 16 章	删除、插入和编辑数据	248	17.6	作业	286
16.1	使用 SqlDataSource 更新、删除 和插入数据	248	17.6.1	测验	286
16.2	修改数据的 SQL 语句	251	17.6.2	答案	286
16.2.1	INSERT 语句	251	17.6.3	练习	286
16.2.2	使用 DELETE 语句删除 数据	252			
16.2.3	使用 UPDATE 编辑 数据	252			
16.3	使用 GridView 编辑和删除 数据	253			
16.3.1	让用户能够删除数据	253			

第 18 章 数据绑定和其他数据相关主题	288
18.1 GridView 和 DetailsView 的字段	288
18.1.1 如何显示 Bit 列	290
18.1.2 使用 HyperLinkField 显示超链接	292
18.1.3 用 ImageField 显示图像	295
18.2 在 WHERE 筛选表达式中使用通配符	296
18.3 数据绑定	297
18.3.1 单向与双向数据绑定的区别	298
18.3.2 为 Web 控件指定数据绑定	298
18.3.3 查看声明标记	301
18.4 总结	302
18.5 问与答	303
18.6 作业	304
18.6.1 测验	304
18.6.2 答案	304
18.6.3 练习	304
第 19 章 使用模板化数据 Web 控件	306
19.1 使用 ListView 控件显示数据	306
19.1.1 ListView 如何显示其 ItemTemplate	307
19.1.2 使用“配置 ListView”选项添加模板	309
19.1.3 手工添加模板	310
19.2 对 ListView 的数据进行分页和排序	312
19.2.1 创建排序界面	313
19.2.2 添加分页功能	314
19.3 使用 FormView 控件每次显示一条记录	317
19.3.1 每次遍历一条记录	318
19.3.2 FormView 的模板	319
19.3.3 定制模板	319
19.4 总结	321

19.5 问与答	321
19.6 作业	322
19.6.1 测验	322
19.6.2 答案	323
19.6.3 练习	323

第四部分 网站导航、用户管理、网页布局、AJAX 和部署

第 20 章 定义网站地图及提供网站导航	326
20.1 ASP.NET 网站导航功能概述	326
20.2 使用网站地图定义网站的结构	327
20.2.1 添加网站地图	328
20.2.2 创建网站地图	329
20.3 使用 SiteMapPath 控件显示面包条	331
20.4 显示完整的网站结构	334
20.4.1 在 TreeView 中显示网站结构	335
20.4.2 使用 Menu 显示网站结构	338
20.5 总结	340
20.6 问与答	340
20.7 作业	341
20.7.1 测验	341
20.7.2 答案	341
20.7.3 练习	342
第 21 章 使用母版页提供网站级网页模板	343
21.1 母版页概述	344
21.2 创建母版页	346
21.3 创建内容页	351
21.4 在母版页中提供默认内容	353
21.5 处理母版页的源代码部分	355
21.6 总结	357
21.7 问与答	358
21.8 作业	358

21.8.1 测验	358	第 23 章 使用 ASP.NET Ajax 创建响应	
21.8.2 答案	358	速度更快的 网页	382
21.8.3 练习	359		
第 22 章 管理网站用户	360	23.1 Ajax 概述	383
22.1 ASP.NET 用户账户概述	360	23.2 使用 ASP.NET Ajax 库	384
22.1.1 配置 ASP.NET 网站以		23.2.1 使用 UpdatePanel 控件	385
支持成员关系	361	23.2.2 使用多个 UpdatePanel	
22.1.2 创建和管理用户	363	控件	388
22.1.3 将用户按角色分类	365	23.2.3 为长时间的部分回传	
22.1.4 创建和管理访问		显示进度消息	390
规则	366	23.3 总结	392
22.1.5 配置网站的 SMTP		23.4 问与答	392
设置	367	23.5 作业	393
22.2 让访客能够新建用户账户	368	23.5.1 测验	393
22.2.1 定制 CreateUserWizard		23.5.2 答案	393
控件	369	23.5.3 练习	394
22.2.2 在用户创建账户后发送		第 24 章 部署网站	395
电子邮件	371	24.1 选择网站托管公司	395
22.2.3 创建非活动用户账户	372	24.2 了解部署过程	397
22.3 使用 Login 控件登录网站	373	24.2.1 创建示例 Web 应用	
22.3.1 定制 Login 控件	374	程序	398
22.3.2 注销	375	24.2.2 上传网站文件	401
22.3.3 指定登录页 URL	376	24.2.3 复制数据库	402
22.4 根据身份验证状态显示		24.2.4 更新 web.config 中的	
内容	376	连接字符串	405
22.5 ASP.NET 网站模板	378	24.3 访问远程网站	407
22.6 总结	379	24.4 总结	408
22.7 问与答	379	24.5 问与答	408
22.8 作业	380	24.6 作业	409
22.8.1 测验	380	24.6.1 测验	409
22.8.2 答案	380	24.6.2 答案	409
22.8.3 练习	381	24.6.3 练习	409

第 1 章

ASP.NET 4 入门

本章将介绍：

- 什么是 ASP.NET；
- 使用 ASP.NET 的系统要求；
- 使用 ASP.NET 前必须安装的软件；
- 安装 .NET Framework、Visual Web Developer 和 SQL Server 2008；
- Visual Web Developer 快速导航；
- 创建一个简单的 ASP.NET 网页并通过 Web 浏览器查看。

ASP.NET 是 Microsoft 倡导的一种令人振奋的 Web 编程技术，它让开发人员能够创建动态网页。动态网页是每次被请求时都将重新生成其内容的网页。例如，登录后，Amazon.com 首页将根据您以前购买的产品显示专门为您推荐的书藉。这是一个动态网页，因为其内容是根据访问该页面的客户专门定制的。在本书中，将介绍如何使用 ASP.NET 创建这样的动态网页。

ASP.NET 是一种健壮而成熟的技术。ASP.NET 1.0 于 2002 年 1 月发布后，很快成为很多人的 Web 编程技术之选。2005 年 11 月，Microsoft 发布了众人期待的 ASP.NET 2.0。两年后的 2007 年 11 月，Microsoft 发布了 ASP.NET 3.5，而 ASP.NET 4 是 2010 年 4 月发布的。

要创建 ASP.NET 网站，需要安装 .NET Framework、Visual Web Developer 和 SQL Server 2008。.NET Framework 是一个用于创建基于 Windows 的应用程序的平台，也是一种用于创建 ASP.NET 网站的底层技术。

Visual Web Developer 是一个创建、编辑和测试 ASP.NET 网站及网页的强大程序。ASP.NET 网页是简单的文本文件，这意味着可以使用任何文件编辑器（如 Microsoft 的“记事本”）对其进行编辑，但如果您以前创建过网站，就知道使用类似 Microsoft Expression Web 和 Adobe Dreamweaver 等工具将使开发过程比使用常规文本编辑器（如“记事本”）更容易。使用 ASP.NET 的目的也如此。

SQL Server 2008 是一个数据库引擎，它是一个专门设计用于高效地存储和查询数据的应用程序。很多网站都与数据库交互，例如，任何电子商务网站都需要显示数据库中的产品信息并在数据库中记录购买信息。从第 13 章开始，将介绍如何通过 ASP.NET 网页创建、查询和修改数据库。

本章重点介绍如何安装必要的软件，以便能够开始创建 ASP.NET Web 应用程序。本章最后将创建一个非常简单的 ASP.NET 网页，但不会深入讨论它。下一章和第 4 章将更深入地介绍 ASP.NET 网页。

1.1 什么是 ASP.NET

您是否曾想知道像 Amazon.com 这种动态网站的后台工作原理是什么？Amazon.com 向顾客显示特定的网页，该网页的内容是根据顾客的偏好和操作动态生成的。例如，如果您在 Amazon.com 有一个账号，当您访问 Amazon.com 主页时，您的名字将显示在顶部，而页面下部将显示个人推荐列表。当您在搜索文本框中输入书名、作者姓名或关键字时，将显示所有匹配的书籍。当您单击特定书名时，它将显示该书的详细信息以及其他用户的评论和评分。当您将该书添加到购物车并结帐时，它将提示您输入信用卡号，然后使用该信用卡收费。

内容根据用户输入或其他信息动态确定的网页称为动态网页。任何网站的搜索引擎页面都是动态网页，因为搜索页面的内容是根据用户输入的搜索条件和 Web 服务器上的文档确定的。另一个例子是 Amazon.com 的个人推荐。您查看页面时，看到的 Amazon.com 推荐的书籍及商品与其他人查看这些页面时看到的不同。具体地说，Amazon.com 推荐的商品是根据您以前查看和购买过的商品动态生成的。

与动态网页相对的是静态网页。静态网页包含的内容不会随访问者或其他外部因素而变化。例如，HTML 页面就是静态网页。请看包含如下标记的 HTML 页面：

```
<html>
<body>
  <b>Hello, World!</b>
</body>
</html>
```

这种页面属于静态网页，因为不管是谁查看该页面或存在哪些外部因素，其输出都相同：粗体“Hello, World!”。仅当有人编辑并保存静态网页时，其内容才会发生改变。

注意：当今几乎所有网站都是静态和动态网页的结合。很难找到只包含静态网页的网站，因为它们的功能非常有限。

**By the
Way**

您需要着重理解的是提供静态网页和提供动态网页的网站之间的差别。

注意：ASP.NET 是众多可用于生成动态网页的技术之一，其他技术包括 ASP（ASP.NET 的前身）、PHP、JSP 和 ColdFusion。如果您使用其他 Web 编程技术开发过 Web 应用程序，将对接下来三小节的内容很熟悉；在这种情况下，请直接跳到“安装 ASP.NET Framework、Visual Web Developer 和 SQL Server 2008”一节。

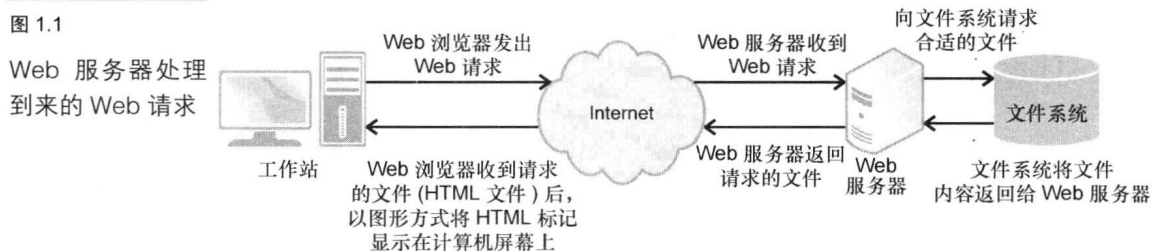
**By the
Way**

1.1.1 提供静态网页

如果您以前开发过网站，就会知道网站需要一个 Web 服务器。

Web 服务器是一种软件应用程序，它不断等待到来的 Web 请求，这些请求都是特定的 URL（参见图 1.1）。Web 服务器检查请求的 URL，找到相应的文件，然后将其发送给发出 Web 请求的客户端。

图 1.1



例如，当您访问 Amazon.com 时，您的浏览器向 Amazon.com 的 Web 服务器发出 Web 请求，即特定的 URL（如/books/index.html）。

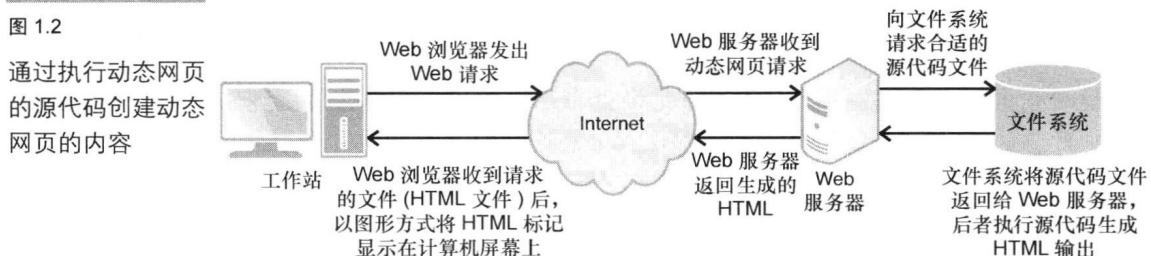
Amazon.com 的 Web 服务器确定 URL 对应的文件，然后将该文件的内容返回给您的浏览器。

这种模型适用于提供静态页面，因为静态页面的内容不会改变。然而，这种简单模型不足以提供动态页面，因为 Web 服务器只是将请求 URL 的对应内容返回给发出请求的浏览器，而不会根据外部输入对请求的 URL 内容进行修改。

1.1.2 提供动态网页

为生成动态内容，动态网页包含它被请求时将执行的源代码（参见图 1.2）。当代码执行时，它生成 HTML 标记，并将结果发送给访问者的浏览器。

图 1.2



使用这种模型时，直到网页被请求时才生成其内容。例如，假设要创建一个显示当前日期和时间的网页。如果使用静态网页来完成这项任务，则需要有人每隔一秒钟编辑一次该网页，从而不停地将页面显示的时间更新为当前时间。这显然不可行。

但使用动态网页时，源代码可获取并显示当前日期和时间。假设某个用户在 2010 年 8 月 1 日下午 4:15:03 访问该页面，当 Web 请求到达时，动态网页的代码将执行，它获取当前日期和时间，并返回给发出请求的 Web 浏览器。该访问者的浏览器将显示该网页的执行时的日期和时间：2010 年 8 月 1 日下午 4:15:03。如果另一个访问者在 7 秒钟后访问该页面，动态网页的代码将再次执行，且返回的日期和时间均为 2010 年 8 月 1 日下午 4:15:10。

实际上，图 1.2 是一个有点过度简化的模型。通常，Web 服务器和执行动态网页源代码的软件是分离的。当 Web 请求到达时，Web 服务器确定请求的页面是静态网页还是动态网页。如果网页是静态的，则直接将其内容发送给发出请求的浏览器（参见图 1.1）。然而，如果请求的网页是动态的（如 ASP.NET 网页），Web 服务器将把执行该页面的任务交给 ASP.NET 引擎（参见图 1.3）。

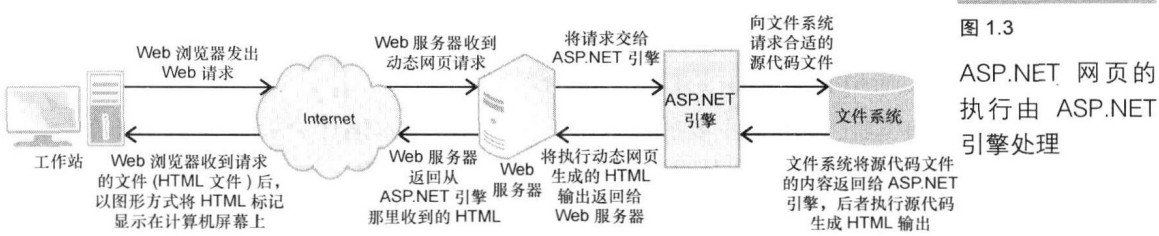


图 1.3
ASP.NET 网页的执行由 ASP.NET 引擎处理

Web 服务器通常可通过请求的文件的扩展名确定请求的页面是动态网页还是静态网页。如果扩展名为 .aspx，Web 服务器便知道请求的是 ASP.NET 网页，因此将请求转交给 ASP.NET 引擎处理。

注意：ASP.NET 引擎是一款软件，它知道如何执行 ASP.NET 网页。其他 Web 编程技术（如 ASP、PHP 和 JSP）也有各自的引擎，它们分别知道如何执行 ASP、PHP 和 JSP 页面。

By the Way

当 ASP.NET 引擎执行 ASP.NET 网页时，引擎以 HTML 输出的形式生成该网页，然后这些 HTML 输出被返回给 Web 服务器，Web 服务器再将其返回给发出 Web 请求的浏览器。

1.1.3 托管 ASP.NET 网页

要查看位于 Web 服务器中的 ASP.NET 网页，需要通过 Web 浏览器请求它。浏览器发送请求给 Web 服务器，后者将请求转发给 ASP.NET 引擎。ASP.NET 引擎处理该页面并将得到的 HTML 返回给 Web 服务器，Web 服务器将其返回浏览器，浏览器再将其显示给用户。开发 ASP.NET 网站时，创建的 ASP.NET 网页保存在个人计算机上。为测试这些页面，您的计算机必须安装 Web 服务器。

幸运的是，您无需担心如何在自己的计算机上安装 Web 服务器。Visual Web Developer（本书将使用该编辑器创建 ASP.NET 网站）包含一个轻量级 Web 服务器，它是为测试 ASP.NET 网页而专门设计的。正如读者在后面将看到的，测试 ASP.NET 网页时，Visual Web Developer 将启动 ASP.NET Development Web Server，然后启动浏览器并发出如下形式的请求：
`http://localhost:portNumber/ASP.NET_Page.aspx。`

请求中的 `http://localhost` 部分告诉浏览器发送请求到当前计算机的 Web 服务器，而不是发送到 Internet 上的其他 Web 服务器。`portNumber` 指定将请求发送到哪个端口。所有 Web 服务器都在特定端口上监听到来的请求。当 ASP.NET Development Web Server 启动时，它选择一个开放端口，该端口将在 URL 的 `portNumber` 部分反映出来。最后，`ASP.NET_Page.aspx` 部分是要测试的 ASP.NET 网页的文件名。

通过 ASP.NET Development Web Server 在本地托管 ASP.NET 网页有如下优点。