



# 新世纪

NEW CENTURY

# ASP.NET动态网页设计

## 应用教程

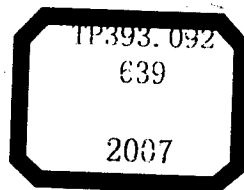
金雪云  
胡海芝 等编著  
郭慧玲



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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



# 新世纪 ASP.NET 动态网页设计 应用教程

金雪云 胡海芝 郭慧玲 等编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书主要介绍了如何在 ASP.NET 中用 C# 语言创建动态 Web 网页的相关技术, 内容涵盖了 ASP.NET 简介、运行 ASP.NET 程序的环境需求、C# 语法基础、使用由 ASP.NET 扩展的 HTML 控件和 Web 服务器控件、ASP.NET 的常用内置对象、.NET 命名空间、在 ASP.NET 中实现事件驱动、在 ASP.NET 程序中使用 Web 增强控件和自定义控件、ADO.NET 和使用 ADO.NET 进行数据库访问、在 ASP.NET 中应用 XML、对 ASP.NET 进行配置和优化以及在 ASP.NET 中实现安全。书中采用了大量实例, 每章都提供测试题和上机实习题, 最后还提供了两套模拟试卷。

本书针对初、中级用户编写, 以实例为主线, 结构清晰, 详略得当, 具有较强的可读性, 可作为各类职业院校 Web 应用程序设计课程的教材, 也可作为各种电脑培训班、辅导班的培训教材。对于希望在 .NET 框架下开发 Web 应用程序的 Web 程序设计人员, 或者希望从基本概念开始学习的 Web 程序爱好者来说也是一本很好的入门级参考书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

### 图书在版编目 (CIP) 数据

新世纪 ASP.NET 动态网页设计应用教程 / 金雪云等编著. —北京: 电子工业出版社, 2007.3

新世纪电脑应用教程

ISBN 978-7-121-03990-4

I. 新… II. 金… III. 主页制作—程序设计—教材 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2007) 第 032679 号

责任编辑: 祁玉芹

印 刷: 北京市天竺颖华印刷厂

装 订: 三河市金马印装有限公司

出版发行: 电子工业出版社出版

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 20.75 字数: 531 千字

印 次: 2007 年 3 月第 1 次印刷

印 数: 6000 册 定价: 28.00 元

凡所购买电子工业出版社图书有缺损问题, 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) 68279077; 邮购电话: (010) 88254888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线: (010) 88258888。

# 出版说明

电脑作为一种工具,已经广泛地应用到现代社会的各个领域,正在改变各行各业的生产方式以及人们的生活方式。在进入新世纪之后,不掌握电脑应用技能就跟不上时代发展的要求,这已成为不争的事实。因此,如何快速、经济地获得使用电脑的知识 and 应用技术,并将所学到的知识和技能应用于现实生活和实际工作中,已成为新世纪每个人迫切需要解决的新问题。

为适应这种需求,各种电脑应用培训班应运而生,目前已成为我国电脑应用技能教育队伍中一支不可忽视的生力军。而随着教育的不断深入,各类高等和中等职业教育中的电脑应用专业也有了长足的发展。然而,目前市场上的电脑图书虽然种类繁多,但适合我国国情的、学与教两相宜的教材却很少。

2001 年推出的《新世纪电脑应用培训教程》丛书,正好满足了这种需求。由于其定位准确、实用性强,受到了读者好评,产生了广泛的影响。但是,三年多来,读者的需求有了提高,培训模式和教学方法都发生了深刻的变化,这就要求我们与时俱进,萃取其精华,推出具有新特色的《新世纪电脑应用教程》丛书。

《新世纪电脑应用教程》丛书是在我们对目前人才市场的需求进行调查分析,以及对高等院校、职业院校及各类培训机构的师生进行广泛调查的基础上,约请长期工作在教学第一线并具有丰富教学与培训经验的教师和相关领域的专家编写的一套系列丛书。

本丛书是为所有从事电脑教学的老师和需要接受电脑应用技能培训或自学的人员编写的,可作为各类高等院校及下属的二级学院、职业院校、成人院校的公修电脑教材,也可用作电脑培训班的培训教材与电脑初、中级用户的自学参考书。它的鲜明的特点就是“就业导向,突出技能,实用性强”。

本丛书并非目前高等教育教材的浓缩和删减,或在较低层次上的重复,亦非软件说明书的翻版,而是为了满足电脑应用和就业现状的需求,对传统电脑教育的强有力的补充。为了实现就业导向的目标,我们认真调研了读者从事的行业或将来可能从事的行业,有针对性地安排内容,专门针对不同行业出版不同版本的教材,尽可能地做到“产教结合”。这样也可以一定程度地克服理论(知识)脱离实际、教学内容游离于应用背景之外的问题,培养适应社会就业需求的“即插即用”型人才。

传统教材以罗列知识点为主，学生跟着教材走，动手少，练习少，其结果是知其然而不知其所以然，举一反三的能力差，实际应用和动手能力差。为了突出技能训练，本丛书在内容安排上，不仅符合“由感性到理性”这一普遍的认知规律，增加了大量的实例、课后的思考练习题和上机实践，使读者能够在实践中理解和积累知识，在知识积累的基础上进行有创造性的实践，而且在内容的组织结构上适应“以学生为中心”的教学模式，强调“学”重于“教”，使教师从知识的传授者、教学的组织领导者转变成为学习过程中的咨询者、指导者和伙伴，充分发挥老师的指导作用和学习者的主观能动性。

为了突出实用性，本丛书采用了项目教学法，以任务驱动的方式安排内容。针对某一具体任务，以“提出需求—设计方案—解决问题”的方式，加强思考与实践环节，真正做到“授人以渔”，使读者在读完一本书后能够独立完成一个较复杂的项目，在千变万化的实际应用中能够从容应对，不被学习难点所困惑，摆脱“读死书”所带来的困境。

本丛书追求语言严谨、通俗、准确，专业词语全书统一，操作步骤明确且采用图文并茂的描述方法，避免晦涩难懂的语言与容易产生歧义的描述。此外，为了方便教学使用，在每本书中每章开头明确地指出本章的教学目标和重点、难点，结尾增加了对本章的小结，既有助于教师抓住重点确定自己的教学计划，又有利于读者自学。

目前本丛书所涉及到的应用领域主要有程序设计、网络管理、数据库的管理与开发、平面与三维设计、网页设计、专业排版、多媒体制作、信息技术与信息安全、电子商务、网站建设、系统管理与维护，以及建筑、机械等电脑应用最为密集的行业。所涉及的软件基本上涵盖了目前的各种经典主流软件与流行面广但技术重要的软件。本丛书对于软件版本的选择原则是：紧跟软件更新步伐，以最近半年新推出的成熟版本为选择的重点；对于兼有中英文版本的软件，尽量舍弃英文版而选用中文版，充分保证图书的技术先进性与应用的普及性。

我们的目标是为所有读者提供读得懂、学得会、用得巧的教学和自学教程，我们期盼着每个阅读本丛书的教师满意、读者成功。

电子工业出版社

# 前 言

从 1996 年 ASP 1.0 诞生到 2001 年, 微软将 ASP 3.0 进行了重大的改进, 并把它加入到 .NET 计划中, 为 Web 应用程序的开发人员提供了高度模块化的、功能更强大的应用程序开发环境。ASP.NET 为 Web 应用程序的开发提供了许多新的特性: 服务器端的代码是经过编译的代码、将服务器端的逻辑与用户的布局信息相分离的代码隐藏技术、可扩展的服务器端的控件、精心设计并且易于使用的数据绑定技术、对于客户端和服务器端身份验证的支持。除此之外, ASP.NET 还提供了语言、工具、库、系统设计和诊断设计的统一。这样, Web 应用程序的开发人员可以摆脱 ASP 环境中的种种束缚, 而使用 .NET 语言进行网页的构建, 使得在 .NET 平台上开发 Web 应用程序与在 .NET 环境中开发其他应用程序没有任何的区别。

## 1. 本书的内容

第 1 章简要介绍了 HTTP 协议、静态网页和动态网页等 Web 基础知识。

第 2 章主要介绍了 C# 的语法基础。

第 3 章中讲解了由 ASP.NET 扩展了的 HTML 控件和 Web 服务器控件, 并着重区分 Web 表单和一般的表单。在这一章中讲述的控件在 ASP.NET 程序设计过程中发挥了重要作用。

第 4 章介绍了 ASP.NET 中常用的内置对象, 包括 Response 对象、Request 对象、Application 对象、Session 对象、Server 对象, 并讲解了环境变量的使用和 Cookie 的使用。所有的这些对象在 ASP 中都存在, ASP.NET 也保留了这些内置对象。

读者通过第 3 章和第 4 章的学习, 可以开始进行基本的 ASP.NET 程序设计, 并可以根据本书配备的例子来深入了解 ASP.NET 程序设计方法。

第 5 章介绍了 ASP.NET 的高级编程方法, 重点讲解了命名空间的概念、什么是类库以及 .NET 中常用类库的使用方法。然后介绍了如何使用 Page 类, 以及如何使用事件驱动的方式进行程序设计, 同时讲解了 ASP.NET 中常用的 Page 事件。

第 6 章介绍了其他的 Web 控件。这些控件包括 Web 增强控件和自定义控件。增强的 Web 控件有日历控件、广告控件以及 6 个验证控件。这些控件是 ASP.NET 新增加的控件, 为程序设计提供了极大的方便。另外, 使用自定义控件可以获得最大的程序设计自由, 并且可以把代码隐藏在 DLL 文件中。

第 7 章介绍了使用 ADO.NET 进行数据库访问的方法。主要讲解了 ADO.NET 相对于 ADO 的优势、ADO.NET 的使用方法以及在数据库应用中常用的 Web 控件 (DataGrid、DataList、Repeater)。与 ADO 相比, 它更容易实现数据共享, 提高了标准化程度并使可编程性大大增强。同时, 从效率上讲, 使用 ADO.NET 将大大提高程序性能。

第 8 章介绍了如何在 ASP.NET 中使用 XML。本章主要讲解了什么是 XML、XML 的格式以及如何在 ASP.NET 中使用 XML 存储和管理数据。

第 9 章介绍了 ASP.NET 的配置和优化方法。本章重点讲解了使用缓存、跟踪和监视 ASP.NET 进程的方法。

第 10 章介绍了在 ASP.NET 中如何实现安全性。本章重点讲解了实现身份验证和授权的几种方法,尤其是基于表单的身份验证方法具有界面美观、实现方便的特点。除此之外,还简单介绍了如何使用 SSL 进行加密。

## 2. 本书适用的读者

本书针对初、中级用户编写,以实例为主线,结构清晰,详略得当,具有较强的可读性,可作为各类职业院校网页程序设计课程的教材,也可作为各种电脑培训班、辅导班的培训教材。对于希望在 .NET 框架下开发 Web 程序的设计人员,或者希望从基本概念开始学习的 Web 程序爱好者来说也是一本很好的入门级参考书。

## 3. 使用本书的要求

运行本书中的所有例子需要建立下面的环境:

- (1) Windows XP 操作系统。
- (2) IIS 5.0 或者 5.0 以上的版本。
- (3) Internet Explorer 6.0 或者更高的版本。
- (4) 本书使用的数据库是 Microsoft SQL Server。
- (5) 在服务器中创建一个文件夹,原样建立例子程序。
- (6) 设置 IIS 中默认 Web 站点的主目录属性页,将它的本地路径设置为刚刚建立的保存了所有例子程序的目录。
- (7) 启动 Web 浏览器,在“地址”栏中输入 <http://localhost/> 后面加入要运行的例子程序的路径和文件名。

本书由金雪云、胡海芝和郭慧玲等编写。在本书的编写过程中,笔者受到来自各方面人士的帮助和支持,在这里表示衷心的感谢。由于时间仓促,本书中难免会有不足之处,敬请各位读者批评指正。

金雪云

## 编辑提示

《新世纪电脑应用教程》丛书自出版以来,受到广大培训学校和读者的普遍好评,我们也收到许多反馈信息。基于读者反馈的信息,为了使这套丛书更好地服务于授课教师的教学,我们为本丛书中新出版的每一本书配备了多媒体教学软件。使用本书作为教材授课的教师,如果需要本书的教学软件,可到网址 [www.tqbook.com](http://www.tqbook.com) 下载。如有问题,可与电子工业出版社天启星文化信息公司联系。

通信地址:北京市海淀区翠微东里甲 2 号为华大厦 3 层 鄂卫华(收)

邮编:100036

E-mail: [qiyuqin@phei.com.cn](mailto:qiyuqin@phei.com.cn)

电话:(010) 68253127 (祁玉芹)

# 目 录

|                                      |    |
|--------------------------------------|----|
| 第 1 章 概述 .....                       | 1  |
| 1.1 Web 基础知识 .....                   | 2  |
| 1.1.1 HTTP 协议 .....                  | 2  |
| 1.1.2 Web 服务器 .....                  | 2  |
| 1.1.3 静态网页 .....                     | 3  |
| 1.1.4 动态网页 .....                     | 4  |
| 1.2 ASP.NET 简介 .....                 | 4  |
| 1.2.1 请求和响应 .....                    | 5  |
| 1.2.2 解释和编译 .....                    | 5  |
| 1.2.3 ASP.NET 的发展历史 .....            | 6  |
| 1.2.4 ASP.NET 的优势 .....              | 6  |
| 1.2.5 ASP.NET 的工作原理 .....            | 7  |
| 1.3 建立 ASP.NET 的运行环境 .....           | 8  |
| 1.3.1 IIS .....                      | 8  |
| 1.3.2 安装 MDAC .....                  | 12 |
| 1.3.3 .NET Framework 和 ASP.NET ..... | 12 |
| 1.4 开始编写 ASP.NET 程序 .....            | 13 |
| 1.4.1 开发环境的选择 .....                  | 13 |
| 1.4.2 编写代码 .....                     | 18 |
| 1.4.3 注释 .....                       | 19 |
| 1.5 本章小结 .....                       | 19 |
| 1.6 习题与上机实习 .....                    | 19 |
| 1.6.1 填空 .....                       | 19 |
| 1.6.2 问答题 .....                      | 20 |
| 1.6.3 上机实习 .....                     | 20 |
| 第 2 章 C#语法基础 .....                   | 21 |
| 2.1 C#的特点和优势 .....                   | 22 |
| 2.2 变量、常量和表达式 .....                  | 22 |
| 2.2.1 变量 .....                       | 22 |



|       |                 |    |
|-------|-----------------|----|
| 2.2.2 | 数据类型 .....      | 24 |
| 2.2.3 | 运算符 .....       | 34 |
| 2.2.4 | 常量 .....        | 40 |
| 2.2.5 | 表达式 .....       | 41 |
| 2.2.6 | 数据类型之间的转换 ..... | 41 |
| 2.3   | 构造类型 .....      | 47 |
| 2.3.1 | 数组 .....        | 47 |
| 2.3.2 | 结构 .....        | 50 |
| 2.3.3 | 枚举 .....        | 52 |
| 2.4   | 控制结构 .....      | 54 |
| 2.4.1 | 顺序 .....        | 54 |
| 2.4.2 | 分支 .....        | 54 |
| 2.4.3 | 循环 .....        | 58 |
| 2.5   | 函数 .....        | 65 |
| 2.5.1 | 函数的定义 .....     | 65 |
| 2.5.2 | 函数的调用 .....     | 66 |
| 2.5.3 | 参数的传递 .....     | 68 |
| 2.5.4 | 变量的作用域 .....    | 72 |
| 2.6   | 习题与上机实习 .....   | 73 |
| 2.6.1 | 填空 .....        | 73 |
| 2.6.2 | 问答题 .....       | 73 |
| 2.6.3 | 上机实习 .....      | 74 |

## 第 3 章 HTML 控件和 Web 服务器控件 ..... 75

|       |                           |    |
|-------|---------------------------|----|
| 3.1   | 表单 .....                  | 75 |
| 3.1.1 | HTML 表单 .....             | 76 |
| 3.1.2 | Web 表单 .....              | 77 |
| 3.2   | HTML 控件 .....             | 78 |
| 3.2.1 | Anchor 控件 .....           | 79 |
| 3.2.2 | Button 控件 .....           | 80 |
| 3.2.3 | InputButton 控件 .....      | 82 |
| 3.2.4 | InputCheckBox 控件 .....    | 82 |
| 3.2.5 | InputRadioButton 控件 ..... | 83 |
| 3.3   | Web 服务器控件 .....           | 84 |
| 3.3.1 | 用于文本输入和显示的内部控件 .....      | 85 |
| 3.3.2 | 用于控制传送的内部控件 .....         | 88 |
| 3.3.3 | 用于选择的列表控件 .....           | 94 |

|                                     |            |
|-------------------------------------|------------|
| 3.4 习题.....                         | 106        |
| 3.4.1 填空 .....                      | 106        |
| 3.4.2 问答题 .....                     | 106        |
| 3.4.3 上机实习 .....                    | 106        |
| <b>第 4 章 ASP.NET 的常用内置对象 .....</b>  | <b>107</b> |
| 4.1 Response 对象.....                | 108        |
| 4.1.1 Response 对象的 Write 方法 .....   | 108        |
| 4.1.2 Response 对象的 Redirect 方法..... | 109        |
| 4.1.3 Response 的 WriteFile 方法.....  | 111        |
| 4.2 Request 对象 .....                | 112        |
| 4.2.1 ServerVariables 属性.....       | 113        |
| 4.2.2 Browser 属性.....               | 115        |
| 4.3 Application 对象 .....            | 116        |
| 4.3.1 如何使用 Application 对象.....      | 116        |
| 4.3.2 Lock 和 UnLock 方法的使用 .....     | 119        |
| 4.3.3 Application 事件.....           | 120        |
| 4.4 Session 对象.....                 | 120        |
| 4.4.1 Session 对象的属性 .....           | 121        |
| 4.4.2 Session 对象的方法 .....           | 122        |
| 4.4.3 Session 对象的事件 .....           | 122        |
| 4.4.4 使用 Session 对象的注意事项 .....      | 122        |
| 4.5 Server 对象 .....                 | 123        |
| 4.5.1 Server 对象的属性 .....            | 123        |
| 4.5.2 Server 对象的方法 .....            | 123        |
| 4.6 Cookie .....                    | 124        |
| 4.6.1 什么是 Cookie .....              | 125        |
| 4.6.2 设置 Cookie.....                | 127        |
| 4.6.3 检索 Cookie.....                | 132        |
| 4.6.4 检测用户是否使用了 Cookie.....         | 133        |
| 4.7 global.asax 文件.....             | 134        |
| 4.7.1 创建 Application 事件代码.....      | 135        |
| 4.7.2 创建 Session 事件代码 .....         | 138        |
| 4.7.3 声明对象 .....                    | 139        |
| 4.8 习题.....                         | 140        |
| 4.8.1 填空 .....                      | 140        |
| 4.8.2 问答题 .....                     | 140        |

|                                       |            |
|---------------------------------------|------------|
| 4.8.3 上机实习 .....                      | 140        |
| <b>第 5 章 ASP.NET 高级编程 .....</b>       | <b>141</b> |
| 5.1 命名空间 .....                        | 142        |
| 5.2 类库 .....                          | 143        |
| 5.2.1 类库和基类 .....                     | 143        |
| 5.2.2 System.Collections 命名空间 .....   | 143        |
| 5.2.3 System.IO 命名空间 .....            | 154        |
| 5.3 Page 类 .....                      | 162        |
| 5.3.1 页面状态 .....                      | 162        |
| 5.3.2 页面指令 .....                      | 164        |
| 5.4 事件驱动 .....                        | 167        |
| 5.4.1 什么是事件驱动 .....                   | 167        |
| 5.4.2 ASP.NET 中的事件 .....              | 168        |
| 5.4.3 ASP.NET 中的服务器控件事件 .....         | 171        |
| 5.5 习题 .....                          | 172        |
| 5.5.1 填空 .....                        | 172        |
| 5.5.2 问答题 .....                       | 172        |
| 5.5.3 上机实习 .....                      | 172        |
| <b>第 6 章 ASP.NET 的其他 Web 控件 .....</b> | <b>173</b> |
| 6.1 日历控件 .....                        | 173        |
| 6.2 广告控件 .....                        | 178        |
| 6.3 验证控件 .....                        | 180        |
| 6.3.1 使用验证控件的注意事项 .....               | 180        |
| 6.3.2 必须字段验证控件 .....                  | 181        |
| 6.3.3 比较验证控件 .....                    | 182        |
| 6.3.4 范围验证控件 .....                    | 184        |
| 6.3.5 正则表达式验证控件 .....                 | 186        |
| 6.3.6 自定义验证控件 .....                   | 187        |
| 6.3.7 验证总结控件 .....                    | 189        |
| 6.4 自定义控件 .....                       | 191        |
| 6.4.1 代码分离技术 .....                    | 191        |
| 6.4.2 编写自定义控件 .....                   | 191        |
| 6.4.3 用户控件 .....                      | 196        |
| 6.5 习题 .....                          | 197        |
| 6.5.1 填空 .....                        | 197        |

|                                    |            |
|------------------------------------|------------|
| 6.5.2 问答题 .....                    | 197        |
| 6.5.3 上机实习 .....                   | 197        |
| <b>第 7 章 访问数据库 .....</b>           | <b>199</b> |
| 7.1 数据库的选择 .....                   | 199        |
| 7.2 ADO.NET 简介 .....               | 200        |
| 7.2.1 ADO.NET 的特点 .....            | 200        |
| 7.2.2 数据访问命名空间 .....               | 200        |
| 7.2.3 Managed Providers .....      | 201        |
| 7.3 用 Connection 对象建立数据库连接 .....   | 202        |
| 7.4 使用 Command 对象执行数据库命令 .....     | 204        |
| 7.5 使用 DataAdapter 对象执行数据库命令 ..... | 208        |
| 7.5.1 DataAdapter 对象 .....         | 209        |
| 7.5.2 DataSet 对象 .....             | 210        |
| 7.6 数据绑定 .....                     | 214        |
| 7.6.1 什么是数据绑定 .....                | 214        |
| 7.6.2 DataGrid 控件 .....            | 215        |
| 7.6.3 DataList 控件 .....            | 226        |
| 7.6.4 Repeater 控件 .....            | 230        |
| 7.7 习题 .....                       | 232        |
| 7.7.1 填空 .....                     | 232        |
| 7.7.2 问答题 .....                    | 232        |
| 7.7.3 上机实习 .....                   | 232        |
| <b>第 8 章 XML 数据管理 .....</b>        | <b>233</b> |
| 8.1 XML 基础 .....                   | 233        |
| 8.2 使用样式表显示 XML .....              | 236        |
| 8.3 在 ASP.NET 中操作 XML 数据 .....     | 239        |
| 8.3.1 写入 XML 数据 .....              | 239        |
| 8.3.2 读取 XML 数据 .....              | 241        |
| 8.3.3 编辑 XML 数据 .....              | 243        |
| 8.3.4 将 XML 转化为字符串 .....           | 244        |
| 8.4 习题 .....                       | 246        |
| 8.4.1 填空 .....                     | 246        |
| 8.4.2 问答题 .....                    | 246        |
| 8.4.3 上机实习 .....                   | 246        |

|                                   |            |
|-----------------------------------|------------|
| <b>第 9 章 ASP.NET 的配置和优化 .....</b> | <b>247</b> |
| 9.1 ASP.NET 的配置 .....             | 247        |
| 9.2 使用缓存 .....                    | 255        |
| 9.3 跟踪 .....                      | 260        |
| 9.4 监视 ASP.NET 进程 .....           | 263        |
| 9.5 习题 .....                      | 264        |
| 9.5.1 填空 .....                    | 264        |
| 9.5.2 问答题 .....                   | 264        |
| <b>第 10 章 ASP.NET 的安全性 .....</b>  | <b>265</b> |
| 10.1 安全性的概念 .....                 | 265        |
| 10.2 Windows 提供的身份验证和授权 .....     | 266        |
| 10.3 IIS 提供的身份验证和授权 .....         | 268        |
| 10.3.1 IP 地址及域名限制 .....           | 268        |
| 10.3.2 启用证书验证用户身份 .....           | 268        |
| 10.3.3 规定验证方法 .....               | 268        |
| 10.4 ASP.NET 提供的身份验证和授权 .....     | 270        |
| 10.4.1 Windows 内置验证 .....         | 271        |
| 10.4.2 基于 Passport 的验证 .....      | 275        |
| 10.4.3 基于表单的验证 .....              | 276        |
| 10.5 SSL 加密方法简介 .....             | 283        |
| 10.6 习题 .....                     | 283        |
| 10.6.1 填空 .....                   | 283        |
| 10.6.2 问答题 .....                  | 284        |
| <b>附录 A 习题答案 .....</b>            | <b>285</b> |
| <b>附录 B 模拟试卷（一） .....</b>         | <b>305</b> |
| <b>附录 C 模拟试卷（二） .....</b>         | <b>311</b> |
| <b>附录 D 模拟试卷（一）答案 .....</b>       | <b>317</b> |
| <b>附录 E 模拟试卷（二）答案 .....</b>       | <b>319</b> |

# 第 1 章

## 概 述

### 教学目标:

ASP.NET是微软继ASP 1.0、ASP 2.0和ASP 3.0之后推出的创建Web应用的环境，Web应用程序的开发人员使用这个开发环境可以实现更加模块化的、功能更强大的应用程序。通过本章的学习，使读者了解Web基础知识，ASP.NET的发展过程，以及ASP.NET的优势。读者通过动手实践，了解如何建立ASP.NET的运行环境，并可以在此环境中运行一个简单的ASP.NET程序。

### 教学重点与难点:

1. Web基础知识。
2. ASP.NET的优势。
3. 建立ASP.NET运行环境，并运行一个简单的程序测试此环境。

1996年，ASP 1.0的诞生给Web应用程序开发带来了福音。早期的Web应用程序开发是十分繁琐的，以至于制作一个简单的动态页面需要编写大量的C代码才能完成。而ASP却允许使用VBScript或JavaScript这样的简单脚本语言，编写嵌入在HTML网页中的代码以进行Web程序设计，并且还可以使用ADO（ActiveX Data Object）组件进行数据库程序设计。

随着时间的推移，人们发现ASP一方面为网站的设计者带来了简便，一方面也使得网站的各种代码难于管理。程序员需要改动一个由大量HTML代码和VBScript、JavaScript代码混合在一起的程序时，他们宁愿写新的代码，也不愿意去改原来的程序，因为原来程序的模块化和可重用性都太低。另外，由于VBScript这样的脚本语言的局限，使得很多功能都不能够轻松地实现，即使实现了也需要写大量的代码。

到了2001年，微软对ASP 3.0进行了重大的改进，Web应用程序的开发人员使用这个开发环境可以实现高度模块化的、功能更强大的应用程序。为了配合.NET计划，微软将其命名为ASP.NET。ASP.NET为Web应用程序的开发提供了许多新的特性。这些特性包括：服务器

端的代码是经过编译的代码、将服务器端的逻辑与用户的布局信息相分离的代码隐藏技术、可扩展的服务器端的控件、精心设计并且易于使用的数据绑定技术、对于客户端和服务器端身份验证的支持。除此之外, ASP.NET 还提供了语言、工具、库、系统设计和诊断计数的统一。这样, Web 应用程序的开发人员可以摆脱 ASP 环境中的种种束缚, 而使用 .NET 语言进行网页的构建, 使得在 .NET 平台上开发 Web 应用程序与在 .NET 环境中开发其他应用程序没有任何的区别。

## 1.1 Web 基础知识

要了解如何利用 ASP.NET 环境建立 Web 应用程序, 必须首先了解一些 Web 基础知识。

### 1.1.1 HTTP 协议

HTTP 协议即超文本传送协议 (Hypertext Transfer Protocol), 是在 Internet 中进行信息传送的协议。浏览器默认使用这个协议。当用户在浏览器的地址栏中输入 URL 字符串 `www.mywebsite.com` 时, 浏览器会自动使用 HTTP 协议来搜索 `http://www.mywebsite.com` 网站的首页。

从浏览器向 Web 服务器发出的搜索某个 Web 网页的请求是 HTTP 请求。当 Web 服务器收到这个请求之后, 会按照请求的要求寻找相应的网页。如果找到, 就把网页的 HTML 代码通过网络传回给发出 HTTP 请求的浏览器; 如果没有找到, 就发送一个错误信息给发出 HTTP 请求的浏览器。响应 HTTP 请求的这些操作称为 HTTP 响应。

HTTP 协议是无状态协议。也就是说, 当使用这种协议时, 所有的请求都是为搜索某一个特定的 Web 网页而发出的。它不知道现在的请求是第一次发出还是已经多次发出, 也不知道这个请求的发送来源。当用户请求一个 Web 网页时, 浏览器会与相关的 Web 服务器相连接, 检索到需要的页面后, 会立即把浏览器与服务器的连接断开。

从程序设计的角度来看, 无状态的特点对于 HTTP 来说是一个缺点, 因为这使得某些功能很难实现。但是由于网络本身的特点, 这是无法改变的事实。因为如果 HTTP 协议是有状态的协议, 就会让一个连接长时间地存在下去, 以判断一个用户到底使用了多长时间, 在这段时间内都做了什么事情。在 Web 环境中, 一个 Web 服务器要保存太多的连接 (因为用户的数量是很难估计的), 从而导致服务器瘫痪。正因如此, 对于所有的 HTTP 请求, Web 服务器都会以同样的方式来对待, 并不会长时间地保存连接。

### 1.1.2 Web 服务器

Web 服务器并不是一台物理的机器。Web 服务器是一种软件, 可以管理各种 Web 文件, 并为提出 HTTP 请求的浏览器提供 HTTP 响应。通常情况下, Web 服务器和浏览器分别处于不同的机器中, 但是在某些特定情况下也可以让它们并存在同一台机器上。例如要学习本书的例子就可以在一台机器上建立一个 Web 服务器, 在它上面编写应用程序, 然后通过同一台机器的浏览器察看这些应用程序的执行结果。

比较常见的 Web 服务器有 Apache 和 IIS (Internet Information Service)。由于 ASP.NET 只能在 IIS 上运行, 所以本书重点介绍 IIS 的相关内容。IIS 是微软公司的操作系统 Windows 2000/XP 所提供的 Web 服务器。在 1.3.1 节中会详细介绍 IIS。

### 1.1.3 静态网页

在动态网页产生之前，所有的网页都是静态的。静态网页就是用纯 HTML 代码编写的网页。这些网页的代码是用一些编辑器输入的，或者是用一些网页设计工具（例如 Macromedia Dreamweaver 等）生成的，保存为.html 或.htm 文件的形式。由于这些网页中没有任何与用户相关的部分，所以在设计完成之后，无论是哪个用户访问这个网页，在什么时候访问这个网页，以何种方式进入这个网页，它的内容都不会有任何不同。

下面是一个静态网页的例子。

例 1.1 (01-01.htm)：一个简单的静态网页，用于显示一个红色的“hello world.”字符串，运行结果如图 1-1 所示。

```
1:  <html>
2:  <body>
3:  <font color=red>hello world.<br></font>
4:  </body>
5:  </html>
```

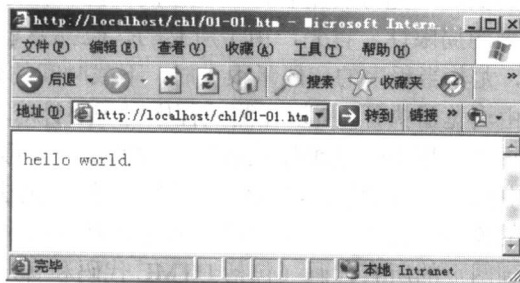


图 1-1 例 1.1 的运行结果

#### 本书约定：

本书中所有的例子都用上面的方式给出。代码前面的数码和冒号并不是代码的内容而是行号，使用它们的目的是为了更清楚地进行代码分析。

例 1.1 是一个最简单的网页，它的目的是显示红色的“Hello World.”字符串。这段代码完全使用 HTML 编写。与 XML 类似，HTML 也是一种置标语言，而且是最简单的一种置标语言，因为在这种语言中，所有标记的含义都已经确定下来，不能有任何改变。例如在这段代码中就使用了 HTML 语言中最常见的一些标记，如<html>、</html>、<body>、</body>、<font>、</font>，等等。如果读者对 HTML 语言不熟悉的话，请参考相关的书籍。使用 HTML 语言只能编写静态网页。因此，只要这个文件（01-01.htm）存在，不论什么用户、在什么时候访问这个网页，以什么方式打开这个网页，都会显示图 1-1 所示的结果。

过多地采用静态网页会导致很大的局限性。如果希望为用户显示一些个性化的信息，使用静态网页就无法达到目的。例如，如果有一个网站的设计人员希望在新年来到的时候在网站首页的最上面显示问候语“新年好！”，而在圣诞节，则在首页的同样位置显示问候语“Merry



Christmas!”, 这些功能在静态网页中是无法实现的。除此之外，静态网页还无法防止用户复制 HTML 代码，因为每个用户都可以采用浏览器的“查看源文件”命令来看到网页的 HTML 代码。

#### 1.1.4 动态网页

动态网页可以为不同的用户提供个性化的服务。对比前面关于静态网页的介绍可以看出，使用动态网页的优势是不言而喻的，而这种动态性是通过程序设计实现的。动态网页的实现方法有两种——客户端程序设计和服务器端程序设计。

客户端程序设计是通过将程序下载到浏览器上来完成所有有关动态特性的。通常的情况是程序员把客户端代码（即可以实现动态内容的程序）编写到 HTML 文件中，当用户提出对这个网页的请求时，这些客户端代码和 HTML 文件的代码一起以响应的方式返回给提出请求的浏览器。由于所有的代码（包括程序和 HTML 标记和客户端代码等）都被浏览器接收，所以这些程序的执行是由浏览器来实现的。目前常见的客户端编程技术有 JavaScript、VBScript 和 Java Applet 等。

在动态网页刚刚出现时，通常是使用客户端程序设计的方法来实现网页的动态服务，因为这样做可以减少服务器的负担，充分利用客户端机器的资源。但是随着技术的发展，客户端程序设计技术体现出很大的局限性。首先，由于所有的代码都要下载到客户端执行，所以网页下载时间会相对增加，尤其当程序的代码量很大时，下载时间的延长会十分明显。其次，由于所有的客户端代码都由浏览器执行，所以在测试程序时需要针对不同的浏览器进行测试，以保证代码的正确执行。这为程序的快速编制设置了很大的障碍。第三，如果需要使用服务器端的资源（例如数据库中的数据），那么采用客户端程序设计就无法实现。第四，采用客户端程序设计无法保证代码的安全，因为所有可以访问到这个网页的用户都可以采用浏览器的“查看源文件”命令来看到网页的所有代码（包括 HTML 代码和客户端程序）。

由于客户端程序设计有这么多的局限性，而现在的服务器的硬件速度又越来越快，相对可以使用的资源也就越来越多，使得采用客户端程序设计以节省服务器端资源需求越来越小，而程序的执行效率、网页响应时间，以及代码的安全性则变得越来越重要，所以服务器端程序设计已经渐渐成为动态网页程序设计的主流。

服务器端程序设计的原理是：程序员编写的代码被保存在服务器上，当用户对某个动态网页提出 HTTP 请求时，这个请求所要访问的网页的代码都在服务器端执行完成，并把执行结果以 HTML 的形式传回浏览器。这样，由于浏览器接收到的只是程序执行的结果，所以上面提到的所有的问题都可以迎刃而解。

## 1.2 ASP.NET 简介

ASP (Active Server Pages) 是一种功能强大而且易于学习的服务器端的脚本编程环境。它是 Microsoft 公司的产品，从 NT Server 操作系统开始就附带这种脚本编程环境。并且，在 NT Workstation、Windows 98 和 Windows 2000 中也都附带这个脚本编程环境。在 2001 年，微软在前面三个版本的 ASP 基础上，推出了全新的 ASP.NET。它开始抛弃前面三个版本都在使用的脚本语言，使用 Visual Basic.NET 或者 C# 作为程序设计语言。但是，无论如何变化，使用这种环境都可以方便地创建动态、快速、交互性强的 Web 站点。