

 高等教育规划教材

Web 应用开发技术 (ASP.NET)

主 编 喻 钧 白小军
副主编 赵宇峰 卢延新



免费提供电子教案

下载网址 <http://www.cmpedu.com>



机械工业出版社
CHINA MACHINE PRESS

高等教育规划教材

Web 应用开发技术(ASP.NET)

主 编 喻 钧 白小军

副主编 赵宇峰 卢延新



机械工业出版社

机械工业出版社

本书从 ASP.NET 初学者的角度出发,对 ASP.NET Web 开发技术进行了由浅入深的详细介绍。全书共分为 12 章,分别介绍了 ASP.NET 程序设计基础、HTML 与 CSS、JavaScript 客户端编程、C#语言基础、ASP.NET 服务器控件、ASP.NET 的对象、数据访问技术、数据绑定、LINQ、Web Service 技术、ASP.NET AJAX 技术等内容,并给出一个完整的综合项目。

书中给出了丰富的实例以帮助读者深入理解和学习,在每章的后面还配有习题和上机练习,本书既适合高等院校的本专科学学生作为教材使用,也适合自学 ASP.NET 动态网页设计的读者作为学习和使用的参考书。

本书配套授课电子课件,需要的教师可登录 www.cmpedu.com 免费注册,审核通过后下载,或联系编辑索取(QQ: 2850823885, 电话: 010-88379739)。

图书在版编目(CIP)数据

Web 应用开发技术: ASP.NET/喻钧, 白小军主编. —北京: 机械工业出版社, 2015. 7

高等教育规划教材

ISBN 978-7-111-51058-1

I. ①W… II. ①喻… ②白… III. ①网页制作工具—程序设计—高等学校—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2015)第 179082 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 王 斌 刘百川 责任编辑: 王 斌

责任校对: 张艳霞 责任印制: 李 洋

三河市国英印务有限公司印刷

2015 年 9 月第 1 版·第 1 次印刷

184mm×260mm·19.5 印张·484 千字

0001-3000 册

标准书号: ISBN 978-7-111-51058-1

定价: 47.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

服务咨询热线:(010)88379833

读者购书热线:(010)88379649

封面无防伪标均为盗版

网络服务

机工官网:www.cmpbook.com

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

金书网:www.golden-book.com

出版说明

当前,我国正处在加快转变经济发展方式、推动产业转型升级的关键时期。为经济转型升级提供高层次人才,是高等院校最重要的历史使命和战略任务之一。高等教育要培养基础性、学术型人才,但更重要的是要加大力度培养多规格、多样化的应用型、复合型人才。

为顺应高等教育迅猛发展的趋势,配合高等院校的教学改革,满足高质量高校教材的迫切需求,机械工业出版社邀请了全国多所高等院校的专家、一线教师及教务部门,通过充分的调研和讨论,针对相关课程的特点,总结教学中的实践经验,组织出版了这套“高等教育规划教材”。

本套教材具有以下特点:

1) 符合高等院校各专业人才的培养目标及课程体系的设置,注重培养学生的应用能力,加大案例篇幅或实训内容,强调知识、能力与素质的综合训练。

2) 针对多数学生的学习特点,采用通俗易懂的方法讲解知识,逻辑性强、层次分明、叙述准确而精炼、图文并茂,使学生可以快速掌握,并学以致用。

3) 凝结一线骨干教师的课程改革和教学研究成果,融合先进的教学理念,在教学内容和方法上做出创新。

4) 为了体现建设“立体化”精品教材的宗旨,本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、源代码或源程序、教学大纲、课程设计和毕业设计指导等资源。

5) 注重教材的实用性、通用性,适合各类高等院校、高等职业学校及相关院校的教学,也可作为各类培训班教材和自学用书。

欢迎教育界的专家和老师提出宝贵的意见和建议。衷心感谢广大教育工作者和读者的支持与帮助!

机械工业出版社

前言

ASP.NET 是在 Microsoft .NET Framework 的基础上构建的、可提供构建企业级 Web 应用程序所需服务的 Web 平台，是创建动态交互网页的强有力的工具。.NET Framework 是用于构建、开发以及运行 Web 应用程序和 Web Service 的公共环境，它主要由三部分组成：编程语言、服务器端和客户端技术、开发环境。ASP.NET 是 .NET Framework 的重要组成部分，它建立在公共语言运行库上，可用于在 Web 服务器上生成功能强大的 Web 应用程序，为 Web 站点创建动态的、交互的 HTML 页面。

作为企业级应用开发的两大主流技术体系之一，.NET 技术近年来发展异常迅速，越来越受到国内外 IT 企业的认可，在各行各业都得到了广泛的应用。基于此，对 .NET 研发人员的需求量也在不断上升，熟悉 .NET 技术体系的学生就业前景很好。

本书作者长期从事 Web 应用开发和 .NET 技术课程的一线教学工作，有着深厚的实践开发经验和丰富的教学经验，熟悉 ASP.NET 和 JavaEE 两大主流技术体系，对面向对象技术、设计模式、软件架构等知识理解较为深刻，能够站在理论的高度来指导实践，同时，作者也非常了解学生的认知规律，进而方便指导教材编写。

Web 应用开发有着很强的技巧性，要求学生从整体上把握软件的架构、框架，合理地使用设计模式，这样才能设计出稳定性好、扩展性强的软件产品。很多培训公司的课程体系和教材注重实践，却缺乏理论深度，培养出来的学生只能够应付就业，却难以取得长远的发展。本书更注重思想方法的培养，将面向对象思想、设计模式和软件架构的知识融入各个章节教学中，尽量使学生知其然并知其所以然，以思想方法指导设计实践。本书紧跟技术潮流，反映了 Web 时代的技术特征，同时没有完全跟风，而是选择主流、稳定的技术进行讲解，保证了内容体系的稳定性。

全书所有程序在 Windows 7、IIS 7.0、.NET Framework 4 构建的环境下测试通过，数据库使用 SQL Server 2012 Express LocalDB 版，开发工具采用 Microsoft Visual Studio 2012 旗舰版。

本书第 1、2、4、5、6 章主要由喻钧编写，第 3、7、8、11 章由白小军编写，第 10、12 章由赵宇峰编写，第 9 章由卢延新编写，曹子建、徐江涛、张永强参与部分章节的编写及校稿工作。全书由喻钧统稿，程序由卢延新等人调试完成。本书电子课件和源程序可以在 www.cmpedu.com 免费下载。

尽管在编写本书的过程中尽了最大努力，但由于编者水平有限，疏漏及不妥之处在所难免，恳请读者批评指正。作者联系邮箱：jyu0117@163.com。

编 者

目 录

出版说明

前言

第1章 ASP.NET 程序设计基础 1

1.1 Web 的工作原理 1

1.1.1 浏览器/服务器模式 1

1.1.2 HTTP 协议 2

1.2 ASP.NET 简介 3

1.2.1 Microsoft .NET Framework 3

1.2.2 ASP.NET 的发展历程 5

1.2.3 ASP.NET 的工作原理 5

1.3 建立 ASP.NET 运行和开发环境 6

1.3.1 安装和配置 IIS 服务器 6

1.3.2 安装 Visual Studio 开发环境 8

1.3.3 安装 SQL Server 数据库系统 11

1.4 实例：创建第一个 ASP.NET Web 站点 11

1.4.1 网站的整体设计 12

1.4.2 网站的部署 12

1.5 习题与上机练习 15

第2章 HTML 与 CSS 16

2.1 使用 HTML 设计网页 16

2.1.1 HTML 文档的基本结构 16

2.1.2 HTML 文档的主要标记 17

2.1.3 HTML 5 新特性 25

2.2 利用 CSS 布局网页 29

2.2.1 在 HTML 中使用 CSS 29

2.2.2 CSS 样式规则 32

2.2.3 CSS 选择器 33

2.2.4 常见的样式属性 37

2.2.5 CSS 3 的新特性 41

2.3 实例：使用 HTML 和 CSS 创建一个用户注册页面 44

2.3.1 设计说明 44

2.3.2 程序实现 44

2.4 习题与上机练习 46

第3章 JavaScript 客户端编程 48

3.1 JavaScript 概述 48

3.1.1	什么是 JavaScript	48
3.1.2	在网页中嵌入 JavaScript 脚本	48
3.1.3	使用 JavaScript 输入与输出信息	50
3.2	JavaScript 基本语法	52
3.2.1	数据类型	52
3.2.2	变量	53
3.2.3	运算符和表达式	53
3.2.4	流程控制	54
3.2.5	函数	58
3.2.6	JavaScript 事件处理	60
3.3	JavaScript 对象编程	61
3.3.1	常用 JavaScript 对象	62
3.3.2	浏览器对象模型	65
3.3.3	文档对象模型	67
3.4	实例：使用 JavaScript 实现客户端数据验证	72
3.4.1	设计说明	72
3.4.2	程序实现	72
3.5	习题与上机练习	74
第4章	C# 语言基础	78
4.1	创建一个简单的 C# 程序	78
4.2	C# 基本语法	80
4.2.1	C#数据类型	80
4.2.2	运算符和表达式	84
4.2.3	程序控制结构	88
4.3	类和对象	92
4.3.1	类和对象的创建	92
4.3.2	属性和方法	96
4.3.3	构造函数和析构函数	97
4.3.4	继承和多态	99
4.4	字符串	100
4.4.1	使用字符串	100
4.4.2	创建动态字符串	105
4.5	集合编程	107
4.5.1	ArrayList	107
4.5.2	哈希表	108
4.5.3	队列	110
4.5.4	堆栈	112
4.6	实例：用 C#动态创建 SQL 数据查询语句	113
4.6.1	设计说明	113
4.6.2	程序实现	113

4.7 习题与上机练习	114
第5章 ASP.NET 服务器控件	116
5.1 ASP.NET 页面的生命周期	116
5.2 服务器控件概述	117
5.2.1 服务器控件的共有属性	117
5.2.2 服务器控件的共有事件	119
5.2.3 服务器控件的分类	119
5.3 标准的 Web 服务器控件	119
5.3.1 文本输入与显示控件	120
5.3.2 控制权转移控件	122
5.3.3 选择控件	124
5.3.4 容器控件	129
5.4 验证控件	131
5.4.1 必须输入验证控件	131
5.4.2 比较验证控件	132
5.4.3 范围验证控件	133
5.4.4 正则表达式验证控件	134
5.4.5 自定义验证控件	136
5.4.6 验证总结控件	137
5.5 用户控件	138
5.5.1 用户控件概述	138
5.5.2 创建用户控件	138
5.5.3 用户控件的使用	140
5.6 实例：商品信息录入及其有效性验证	141
5.6.1 设计说明	141
5.6.2 程序实现	142
5.7 习题与上机练习	144
第6章 ASP.NET 的对象	145
6.1 HTTP 请求处理	145
6.1.1 Response 对象	145
6.1.2 Request 对象	147
6.1.3 Server 对象	150
6.2 状态信息保存	154
6.2.1 Application 对象	154
6.2.2 Session 对象	158
6.2.3 Cookie 对象	160
6.2.4 ViewState 对象	162
6.3 实例：一个简单的购物车	163
6.3.1 设计说明	163
6.3.2 程序实现	164

6.4 习题与上机练习	167
第7章 数据访问技术	169
7.1 ADO.NET 体系结构	169
7.1.1 ADO.NET 数据提供程序	169
7.1.2 ADO.NET 数据集	170
7.1.3 ADO.NET 类的组织	171
7.2 使用基于连接的对象访问数据库	172
7.2.1 访问数据库的一般方法	172
7.2.2 使用 Connection 对象	174
7.2.3 使用 Command 对象	178
7.2.4 使用 DataReader 对象	185
7.3 使用基于内容的对象访问数据库	186
7.3.1 使用 DataTable 对象	186
7.3.2 使用 DataView 对象	188
7.3.3 使用 DataAdapter 对象	190
7.4 实例：分类浏览商品信息的页面	192
7.4.1 设计说明	192
7.4.2 程序实现	192
7.5 习题与上机练习	193
第8章 数据绑定	195
8.1 数据绑定基础	195
8.1.1 数据绑定表达式	195
8.1.2 单值绑定	196
8.1.3 重复值绑定	197
8.2 数据源控件	199
8.2.1 数据源控件概述	199
8.2.2 SqlDataSource 控件	202
8.2.3 ObjectDataSource 控件	205
8.3 数据绑定控件	208
8.3.1 GridView 控件	208
8.3.2 ListView 控件	227
8.3.3 DetailsView 控件	229
8.3.4 FormView 控件	231
8.4 实例：使用数据绑定控件的分类浏览商品信息页面	234
8.4.1 设计说明	234
8.4.2 程序实现	234
8.5 习题与上机练习	236
第9章 LINQ	238
9.1 LINQ 概述	238
9.1.1 LINQ 的体系结构	239

9.1.2 LINQ 与 ADO.NET 的关系	239
9.2 使用 LINQ 查询	240
9.2.1 from 子句	240
9.2.2 where 子句	241
9.2.3 select 子句	242
9.2.4 group by 子句	242
9.2.5 orderby 子句	243
9.3 使用 LINQ 对数据库进行操作	243
9.3.1 创建数据源	243
9.3.2 数据库的查询	245
9.3.3 数据库的插入	246
9.3.4 数据库的更新	247
9.3.5 数据库的删除	247
9.4 LINQ 中的数据绑定	248
9.4.1 LinqDataSource	248
9.4.2 数据的绑定	249
9.5 实例: 利用 LINQ 更新商品信息	251
9.5.1 设计说明	251
9.5.2 程序实现	251
9.6 习题与上机练习	252
第10章 Web Service 技术	253
10.1 Web Service 的概念	253
10.1.1 Web Service 的定义和概念	253
10.1.2 Web Service 的基本特征	254
10.1.3 Web Service 的优势	254
10.2 Web Service 的实现技术	255
10.2.1 Web Service 的体系结构	255
10.2.2 Web Service 的协议栈	256
10.2.3 Web Service 的核心元素	256
10.3 构建 ASP.NET Web Service	258
10.3.1 使用 Visual Studio 创建 Web Service	258
10.3.2 测试 Web Service	259
10.3.3 发布 Web Service	260
10.4 使用 Web Service	261
10.4.1 添加 Web 引用	261
10.4.2 访问 Web Service	263
10.5 实例: 使用 Web Service 实现商品分类查询	263
10.5.1 设计说明	263
10.5.2 程序实现	263
10.6 习题与上机练习	265

第 11 章 ASP.NET AJAX	267
11.1 AJAX 技术基础	267
11.2 传统的 AJAX 编程方式	268
11.3 使用 Microsoft AJAX 技术	273
11.3.1 Microsoft AJAX 概述	273
11.3.2 UpdatePanel 控件	274
11.3.3 UpdateProgress 控件	277
11.3.4 Timer 控件与定时刷新	278
11.4 使用 AJAX Control Toolkit	279
11.4.1 安装 AJAX Control Toolkit	279
11.4.2 使用 AJAX Control Toolkit 中的控件	281
11.5 实例：一个抢购倒计时的 AJAX 应用	283
11.5.1 设计说明	283
11.5.2 程序实现	284
11.6 习题与上机练习	285
第 12 章 项目案例：网上书店系统	286
12.1 系统需求分析	286
12.2 系统设计	287
12.2.1 功能模块设计	287
12.2.2 数据库设计	287
12.3 页面设计	290
12.4 系统实现	293
12.4.1 系统主页面	293
12.4.2 图书检索模块	295
12.4.3 用户购书模块	296
12.4.4 提交订单模块	297
12.4.5 后台管理模块	298
12.5 系统的发布	300
参考文献	302

第 1 章 ASP.NET 程序设计基础

Web 技术的迅速发展极大地改变了整个世界，也深刻地影响了人们的日常生活。作为一个出色的 Web 开发框架，ASP.NET 已经成为 Web 应用开发中不可缺少的中坚力量，由此衍生出数以万计的产品、服务以及开源工程。认知 ASP.NET 程序设计基础将有助于理解 Web 应用

的开发。本章将从 Web 的工作原理开始，阐述 Web 开发基础知识，并与读者一起搭建 ASP.NET 的运行和开发环境，最终创建第一个 ASP.NET Web 站点。

1.1 Web 的工作原理

当用户访问 Web 站点时，数据是遵从 HTTP 协议进行传输的。HTTP 即超文本传输协议，是基于浏览器/服务器（Browser/Server，B/S）模式，使用 TCP 连接在应用层进行可靠的数据传输。

1.1.1 浏览器/服务器模式

B/S 模式是随着 Web 技术发展而逐渐成熟的软件系统体系结构。在这种结构中，所有的应用程序以及数据库系统都安装在服务器（Server）上，客户端只需安装任意一个浏览器（Browser）即可，它是零维护的。用户通过浏览器向服务器发出一个请求（例如，在地址栏中输入一个网址，单击超链接或按钮），服务器处理该请求后，将结果以 HTML 的形式返回给客户端，如图 1-1 所示。B/S 结构采用 Internet/Intranet 技术，适用于广域网环境。它可以根据访问量动态地配置 Web 服务器、应用服务器，以支持更多的客户。其优点是代码可重用性好，系统的扩展维护简单。



图 1-1 B/S 软件体系结构

与 B/S 模式对应的是 C/S（Client/Server，客户机/服务器）模式，C/S 模式是较早的软件系统体系结构，主要用于局域网环境，其服务器通常采用高性能 PC 或工作站，客户机需要安装专门的客户端软件，即应用程序。这种结构能充分发挥客户端 PC 的处理能力，客户端响应速度快，但系统的可扩展性和可维护性差。任意一台客户端的应用程序出现问题，都将导致系

统不能正常工作，需要重新安装和维护。另外，由于采用 Intranet 技术，适用于局域网环境的可连接用户数量有限，当用户数量增多时，系统性能会明显下降，代码的可重用性差。

B/S 结构和 C/S 结构的比较如表 1-1 所示。

表 1-1 B/S 结构与 C/S 结构的比较

	B/S 软件体系结构	C/S 软件体系结构
硬件环境	广域网，不必是专门的网络环境，只要是能接入 Internet 的用户均可	局域网，专门的小范围网络硬件环境，用户固定，用户数量有限
系统维护	客户端零维护，易于实现系统的无缝升级	升级和维护难，成本高
软件重用性	多重结构，各构件相对独立，可重用性较好	单一结构，软件整体性较强，各部分间的耦合性强，可重用性较差
平台相关性	客户端和服务端是平台无关的	客户端和服务端是平台相关的，多是 Windows 平台
安全性	面向不可知的用户群，对信息安全的控制能力相对较弱	面向相对固定的用户群，对信息安全的控制能力强

1.1.2 HTTP 协议

HTTP 协议定义了 Web 浏览器和 Web 服务器之间交换数据的过程以及数据本身的格式，是客户机与服务器交互遵守的协议。

HTTP 协议的工作原理如图 1-2 所示。它表示基于 HTTP 的信息交换过程，共分为 4 个步骤：建立连接、发送请求信息、返回响应信息、断开连接。首先，客户端的浏览器向服务器的某个端口发出请求，建立与服务器的连接，通常默认端口号为 80；在连接建立后，客户端向服务器发出一个请求（Request）；服务器接收和处理请求后，返回一个响应（Response）页面；然后，客户端与服务器之间的连接断开，通信结束。一般来说，任何一方都可结束连接，但通常是由客户端收到所请求的信息后关闭连接。

HTTP 协议是一种请求/应答协议，它通过客户机和服务器之间发送请求、应答消息的方式工作。使用 HTTP 消息头（header），可以实现客户机与服务器之间的 HTTP 请求和 HTTP 应答。

一个 HTTP 请求消息包括三部分：一个请求行、若干消息头，以及消息实体内容。其中一些消息头和实体内容是可选的，消息头和实体内容之间用空行隔开。

例如，图 1-3 所示的矩形框就是一个 HTTP 请求消息的内容。第 1 行为请求行，中间的若干行为消息头，空行表示消息头的结束，最后是消息的实体内容。

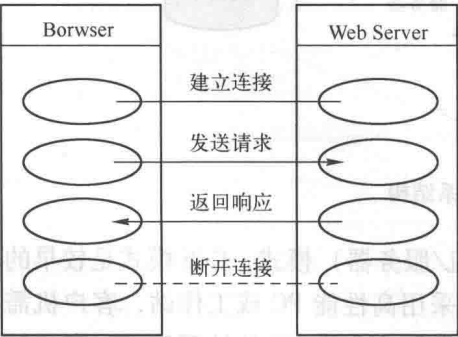


图 1-2 HTTP 的工作原理

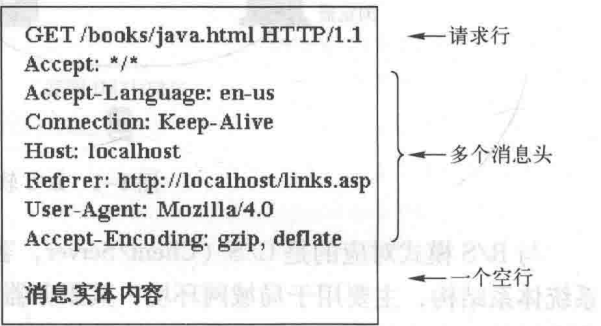


图 1-3 HTTP 请求消息

在图 1-3 中，第 1 行（请求行）表示客户端采用 GET 方式向服务器传送数据，请求的 URL 地址为本地文件/books/java.html，使用的 HTTP 协议版本为 1.1。

从第 2 行开始，直到空行前是 HTTP 请求头域的内容。它的常用参数如下：

- Accept：用于指定客户端可以接受的 MIME 类型，如 */* 表示任何类型。
- Accept-Charset：指定客户端可以使用的字符集，如 UTF-8。
- Accept-Language：指定客户端的接收语言，可以指定多个，如 zh-cn。
- Accept-Encoding：指定客户端的接收编码，如 gzip、deflate。
- Connection：指示处理完本次请求/响应后，客户端与服务器是否继续保持连接，取值为 Keep-Alive 或 close。默认为 Keep-Alive。
- Host：指定资源所在的主机和端口号。
- Referer：确定获得请求 URI 的资源地址。
- User-Agent：用户代理，指定初始化请求的客户端程序，如浏览器等。

同请求消息类似，一个 HTTP 响应消息包括三部分：一个状态行、若干消息头，以及实体内容。其中一些消息头和实体内容是可选的，消息头和实体内容之间用空行隔开。

例如，图 1-4 所示为一个 HTTP 响应消息的内容。第 1 行为状态行，中间为若干消息头，空行表示消息头的结束，最后是消息的实体内容。

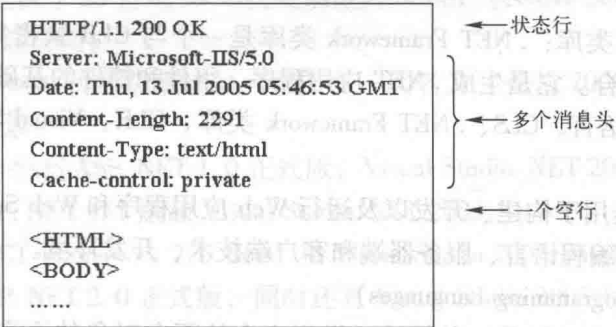


图 1-4 HTTP 响应消息

1.2 ASP.NET 简介

ASP.NET 是在 Microsoft .NET 框架的基础上构建的，可提供构建企业级 Web 应用程序所需服务的一个 Web 平台。ASP.NET 与 Microsoft .NET 框架平台紧密结合是 ASP.NET 的最大特点。

1.2.1 Microsoft .NET Framework

.NET 是一种面向网络、支持各种用户终端的开发平台环境。在 .NET 平台上，不同网站间通过相关协定联系在一起，形成自动交流、协同工作的模式，为系统提供全面的服务。

.NET Framework 是一个集成在 Windows 中的组件，它是为其运行的应用程序提供各种服务的托管执行环境，如图 1-5 所示，它主要包括两个组件：公共语言运行时（Common Language Runtime, CLR）和 .NET Framework 类库（.NET Framework Class Library）。

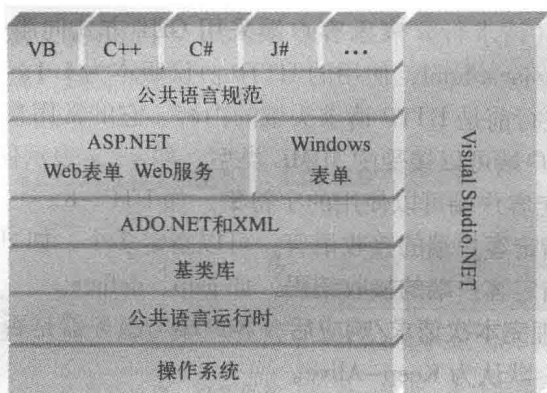


图 1-5 .NET Framework

- **公共语言运行时 (CLR)**: CLR 是 .NET Framework 的基础。它为执行 .NET 脚本语言编写的代码提供了一个运行环境。CLR 管理 .NET 代码的执行, 提供内存管理、线程管理、代码执行、安全验证、远程处理等服务, 并保证应用和底层操作系统之间必要的分离。同时, CLR 使得开发人员可以调试和进行异常处理。要执行这些任务, 需要遵循公共语言规范 (Common Language Specification, CLS)。CLS 描述了运行库能够支持的数据类型的子集。
- **.NET Framework 类库**: .NET Framework 类库是一个与 CLR 紧密集成的、面向对象的、可重用的类型集合。它是生成 .NET 应用程序、组件和控件的基础, 如图 1-5 所示, 它包括 .NET 脚本语言、CLS、.NET Framework 类库、CLR、Visual Studio .NET 集成开发环境等。
- **.NET Framework** 是用于构建、开发以及运行 Web 应用程序和 Web Service 的公共环境。它主要是由三部分组成: 编程语言、服务器端和客户端技术、开发环境。

(1) 编程语言 (Programming Languages)

- **C# (读作 C sharp)**: C# 是一种简洁、类型安全的面向对象的语言, 它是 Microsoft 公司专门为 .NET 量身定做、为生成在 .NET Framework 上运行的应用程序而设计的。C# 从 C 和 C++ 衍生而来, 更像 JAVA。使用 C# 可以创建 XML Web Services、分布式组件、客户端/服务器应用程序、数据库应用程序等。
- **Visual Basic (简称 VB .NET)**: Visual Basic .NET 是从 Visual Basic 语言演变而来的, 面向 .NET Framework, 能生成类型安全和面向对象应用程序的一种语言。VB .NET 是 Visual Studio .NET 的一部分, 是一套完整的、可生成企业级 Web 应用程序的开发工具。
- **J# (读作 J sharp)**: J# 是一种供 Java 程序员构建在 .NET Framework 上运行的应用程序和服务的语言。

(2) 服务器端和客户端技术 (Server Technologies and Client Technologies)

- **ASP .NET (Active Server Pages)**: ASP .NET 是建立在 .NET Framework 之上, 利用 CLR 在服务器端为用户提供建立企业级 Web 应用服务的编程框架。ASP .NET 主要包括 Web Form 和 Web Service 两种编程模型。前者提供建立功能强大的、基于 Form 的可编程 Web 页面。后者提供在异构网络环境下获取远程服务、连接远程设备、交互远程应用的编程界面。
- **Windows Forms (Windows Desktop Solutions)**: Windows Forms 是 .NET Framework 的智能

客户端组件，用于创建应用程序和用户界面。Windows Forms 应用程序是基于 System.Windows.Forms 命名空间中的类；通过在窗体上放置控件并对用户操作（如鼠标单击）进行响应来构建。

- Compact Framework (PDA/Mobile Solutions): Compact Framework 是为了在移动设备和嵌入式设备上运行而设计的。它包含 .NET Framework 中的类库的子集，同时还包含一些专有类。

(3) 开发环境 (Development Environments)

- Visual Studio: Visual Studio 是一个完整的集成开发环境 (Integrated Development Environment, IDE)，用于生成 ASP.NET Web 应用程序、XML Web Services、桌面应用程序和移动应用程序。VB.NET、Visual C++ .NET、Visual C# .NET 和 Visual J# .NET 全都使用相同的 IDE，该环境允许它们共享工具并有助于创建混合语言解决方案。
- Visual Web Developer: Visual Web Developer 是一个功能齐备的开发环境，可用于创建 ASP.NET Web 应用程序。Visual Web Developer 提供网页设计、代码编辑、测试和调试，以及将 Web 应用程序部署到承载服务器等功能。

1.2.2 ASP.NET 的发展历程

微软的 Web 开发起源于 20 世纪 90 年代晚期经典的 ASP (Active Server Pages)。开发者可以利用 VB 语言编写动态页面。

2000 年 6 月，Microsoft 公司总裁比尔·盖茨在一次名为“论坛 2000”的会议上发表了演讲，描绘了 .NET 的美景。

2002 年，Microsoft 公布 ASP.NET 1.0 正式版，Visual Studio .NET 2002 也同步发行。

2003 年，推出 ASP.NET 1.1 版和 Visual Studio .NET 2003。

2004 年，发布 .NET Framework 2.0 Beta1 和 Visual Studio 2005 Beta1。

2005 年，推出 ASP.NET 2.0 正式版，同时还有 Visual Studio 2005 和 SQL Server 2005 正式版。ASP.NET 2.0 的发布是 .NET 技术走向成熟的标志。

2007 年，Microsoft 推出了 ASP.NET 3.0 版本，即 .NET Framework 3.0。ASP.NET 3.0 就是在 2.0 的基础上扩展了 WPF、WCF 和 WWF。

2007 年 11 月，推出 .NET Framework 3.5 和 Visual Studio 2008 正式版，加入了 LINQ（语言集成查询）并支持 ASP.NET AJAX，Web 开发技术又前进了一大步。

2010 年 4 月，发布 Visual Studio 2010 正式版，同时推出 .NET Framework 4.0、Visual Studio 2010 CTP，并且支持开发面向 Windows 7 的应用程序。除了 Microsoft SQL Server，它还支持 IBM DB2 和 Oracle 数据库。

2012 年，ASP.NET 4.5 与 Visual Studio 2012 正式发布。这个版本包括 ASP.NET MVC 4.5 以及全新的 ASP.NET Web API，这使得编写基于 REST 的 Web 服务更加容易。随后 Microsoft 在开源协议（Apache 协议 2.0）下发布了 Web 栈（Web Stack），其中包括 ASP.NET MVC，Razor，以及 ASP.NET Web API。

1.2.3 ASP.NET 的工作原理

ASP.NET 的工作原理如图 1-6 所示。

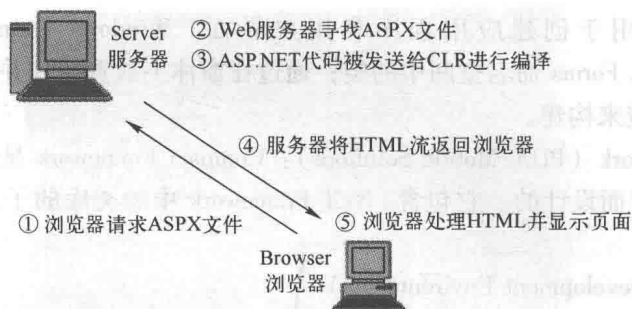


图 1-6 首次请求 ASP. NET 页面的处理过程

- 1) Web 浏览器发送一个 HTTP 请求到 Web 服务器，要求访问一个 Web 网页。
- 2) Web 服务器分析这个 HTTP 请求，定位所请求的 Web 网页的位置。
- 3) 如果请求的网页是一个 HTML 文件，则服务器直接返回该文件。如果请求的是 ASP. NET 文件，那么 IIS 就把该文件传送到 `aspnet_isapi.dll` 进行处理，后者把 ASP. NET 代码提交给 CLR。若是首次请求该 ASP. NET 文件，就由 CLR 编译并执行，得到纯 HTML 结果；若是已经执行过这个文件，就直接执行编译好的程序并得到纯 HTML 结果。
- 4) 最后把从 3) 中得到的 HTML 文件传回浏览器作为 HTTP 响应。
- 5) 浏览器收到这个 HTTP 响应之后，就可以显示 Web 网页。

1.3 建立 ASP. NET 运行和开发环境

要运行 Web 程序，必须首先建立一个 Web 服务器，然后从任意一台 Web 浏览器访问该服务器上的 Web 程序。建立 ASP. NET 的运行环境，需要安装 Web 服务器 IIS 和 .NET Framework。

要运行 ASP. NET 程序，首先应在服务器上设置网站的根目录，并将网页文件放在该目录下，然后通过 IIS 管理器建立一个指向该网站根目录的虚拟目录，最后，在浏览器地址栏中输入相应的地址（通常包含虚拟目录）运行 ASP. NET 程序即可。

要编写和调试 ASP. NET 程序，通常选择 Visual Studio 集成开发环境。本书以当前主流配置的 Visual Studio 2012 和 SQL Server 2008 R2 为主要开发环境进行讲解。

1.3.1 安装和配置 IIS 服务器

IIS (Internet Information Server, 互联网信息服务) 是 ASP. NET 唯一可以使用的 Web 服务器，目前常用的版本是 IIS 7。下面将以 Windows 7 为例，简述 IIS 7 的安装和配置过程。

1. 安装 IIS

- 在“控制面板”中，选择“打开或关闭 Windows 功能”。这是一个触发 UAC 的操作，如果 Windows 7 中没有关闭 UAC，则会弹出提示信息，确认并继续；
- 如果仅需要 IIS 7.0 支持静态内容，可直接选中“Internet 信息服务”，如果希望支持动态内容，则需展开“万维网服务”分支，将所需的选项全部选中；
- 单击“确定”按钮，Windows 7 即启动 IIS 的安装过程；
- 安装完成后，打开浏览器输入“`http://localhost/`”，检查 IIS 是否正常。如果出现 IIS 的信息页面，则表示 IIS 安装成功，如图 1-7 所示。此时，通常会在硬盘 C 上自动创建文