



普通高等教育“十一五”国家级规划教材

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

# JavaScript程序设计 基础教程 (第2版)

Beginning Web Programming in JavaScript (2nd Edition)

阮文江 编著

- 简单实用，突出引导
- 内容全面，注重基础
- 实例丰富，趣味性强



精品系列



人民邮电出版社  
POSTS & TELECOM PRESS

TP312.14



普通高等教育“十一五”国家级规划教材

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

# JavaScript程序设计 基础教程 (第2版)

Beginning Web Programming in JavaScript (2nd Edition)

阮文江 编著



精品系列

人民邮电出版社

北京

## 图书在版编目 (CIP) 数据

JavaScript程序设计基础教程 / 阮文江编著. -- 2  
版. -- 北京: 人民邮电出版社, 2010. 8  
21世纪高等学校计算机规划教材  
ISBN 978-7-115-23084-3

I. ①J… II. ①阮… III. ①JAVA语言—程序设计—  
高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第105557号

## 内 容 提 要

本书是学习 JavaScript 动态网页编程技术的基础教材, 共分 10 章, 主要内容包括: Web 技术概述、HTML/XHTML 制作、层叠样式表 (CSS) 技术、JavaScript 编程基础、基本流程控制、函数、对象编程、浏览器对象和 HTML DOM、事件驱动编程和 JavaScript 网页特效等。为便于教学, 每章均附有练习题。

本书内容丰富, 讲解循序渐进、深入浅出, 简明易懂。本书可作为高等院校本、专科各专业 JavaScript 程序设计、动态网页制作、大学计算机基础 (编程入门部分) 等课程的教材, 也可用作电子商务、电子政务的辅助培训教材。

21 世纪高等学校计算机规划教材

### JavaScript 程序设计基础教程 (第 2 版)

- 
- ◆ 编 著 阮文江  
责任编辑 滑 玉  
执行编辑 董 楠
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
中国铁道出版社印刷厂印刷
  - ◆ 开本: 787×1092 1/16  
印张: 18.5 2010 年 8 月第 2 版  
字数: 480 千字 2010 年 8 月北京第 1 次印刷

---

ISBN 978-7-115-23084-3

定价: 32.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

# 前 言

---

---

---

---

---

---

---

---

本书是动态网页编程的入门级教材，适用于 JavaScript 程序设计、动态网页设计等基础课程教学。本书主要面向 Web 技术的初学者，立足于以下特色。

- 起点低、面向初学者：本书适用于那些不懂编程、只具备计算机基本使用能力的读者，如高校低年级学生和相应层次的自学者。
- 内容全面、实用性强：本书较为全面地介绍了 JavaScript 的基础程序设计技术，并且在选材上侧重实用性技术和案例。
- 注重基础性、简单性：作为学习 Web 技术和程序设计技术的入门级教材，本书着重介绍 JavaScript 的基础知识和技术，并且强调简单性，有意淡化或忽略 JavaScript 的复杂技术，只提供短小、实用的程序示例，以培养、增强初学者学习 JavaScript 技术的信心。
- 突出引导：本书注重培养读者的自主学习能力，叙述上力求深入浅出、简明易懂、突出引导。特别是强调引导读者善用软件开发工具（如设计工具、调试器、帮助系统等），使读者在掌握基本的 JavaScript 实用技术的基础上，逐步具备进一步学习 JavaScript 高级技术的能力。
- 实例丰富、趣味性强：本书几乎为每个知识点设计了典型程序示例，并且结合动态网页技术为读者提供大量有趣的动态网页编程实例，以提高读者学习 JavaScript 编程技术的兴趣。

本书的上一版本是 2004 年出版的，出版后受到读者的普遍欢迎。本次改编的目的是为了适应 Web 技术的发展，并解决 2004 版过于简单的问题，主要改进之处在于以下两个方面。

- 基于 Web 标准化理念，重编了第 2 章“HTML/XHTML 制作”和第 3 章“层叠样式表 (CSS) 技术”等章节，并且使用 XHTML 语言编制所有示例的页面代码。
- 适当提高本书的难度。也就是增加介绍 JavaScript 自定义对象技术，并且较为系统地介绍 HTML DOM 技术和事件驱动编程技术。

为了便于教师和一般读者使用，编者为本书准备了教学辅助材料，包括各章的电子讲稿、例题文件以及习题答案。读者可以在人民邮电出版社教学服务与资源网网站（[www.ptpedu.com.cn](http://www.ptpedu.com.cn)）的“下载区”中下载这些资料。

在本书的编写过程中，得到中山大学计算机科学系公共计算机教研室各位教师和其他相关教师的支持和帮助，在此表示衷心感谢。

由于时间紧迫以及 Web 技术发展迅速，书中难免有不足之处，恳请读者批评指正。

编者电子邮件地址：[pusrwj@mail.sysu.edu.cn](mailto:pusrwj@mail.sysu.edu.cn)

编 者  
2010 年 3 月

# 目 录

<b>第 1 章 Web 技术概述</b> .....1	2.1.4 基本结构标签.....16
1.1 Internet 简述.....1	2.1.5 HTML 文档的良好性.....16
1.1.1 Internet 定义.....1	2.2 文档分段.....17
1.1.2 IP 地址与域名.....1	2.2.1 标题.....17
1.1.3 TCP/IP.....2	2.2.2 水平线.....18
1.2 Web 基本知识.....2	2.2.3 段落.....19
1.2.1 Web 定义.....2	2.2.4 强制分行.....20
1.2.2 HTTP.....3	2.3 标记行内元素.....21
1.2.3 HTML 文档.....3	2.3.1 块级元素与行内元素.....21
1.2.4 网站.....4	2.3.2 标记短语.....22
1.2.5 网址.....4	2.3.3 标记计算机代码.....24
1.3 Web 浏览器与 Web 服务器.....4	2.3.4 标记指定格式的文本.....24
1.3.1 Web 浏览器.....4	2.4 建立超链接.....25
1.3.2 Web 服务器程序.....6	2.4.1 文本链接.....26
1.4 动态网页及相关技术.....6	2.4.2 图像链接.....26
1.4.1 客户端动态网页.....6	2.4.3 锚点链接.....26
1.4.2 服务器端动态网页.....7	2.4.4 URL 的多种形式.....27
1.5 Web 标准化.....7	2.5 制作列表.....28
1.5.1 Web 标准.....7	2.5.1 常规列表.....28
1.5.2 制作符合 Web 标准的网页.....8	2.5.2 列表嵌套.....29
1.6 网页制作工具.....9	2.5.3 定义列表.....30
1.6.1 网页制作方式.....9	2.6 制作表单.....31
1.6.2 FrontPage 2003 与 SharePoint Designer 2007.....9	2.6.1 定义表单.....31
1.6.3 Dreamweaver CS4.....10	2.6.2 定义表单控件.....32
习题.....11	2.7 添加多媒体.....35
<b>第 2 章 HTML/XHTML 制作</b> .....13	2.8 通用属性.....36
2.1 基本概念.....13	2.9 扩展 HTML 标签.....37
2.1.1 XML 与 XHTML.....13	2.10 <meta>标签.....38
2.1.2 HTML 标签.....13	2.11 配置 Web 服务器.....39
2.1.3 使用专业工具制作 HTML 页面.....14	2.11.1 安装 IIS.....39
	2.11.2 配置 Web 站点.....40
	习题.....41

<b>第 3 章 层叠样式表技术</b> .....46	<b>3.9 列表样式</b> .....75
3.1 CSS 简介.....46	3.9.1 CSS 列表属性.....75
3.2 定义样式.....47	3.9.2 内容生成属性.....76
3.2.1 样式定义格式.....47	3.9.3 自定义编号.....78
3.2.2 基本选择器.....48	3.9.4 多级编号.....78
3.3 使用样式.....49	3.10 表格制作.....81
3.3.1 嵌入样式表.....50	3.10.1 制作常规表格.....81
3.3.2 链接外部样式表.....50	3.10.2 表格行分组.....83
3.3.3 内嵌样式.....51	3.10.3 将其他元素显示为表格.....84
3.4 CSS 基本格式化属性.....52	3.10.4 CSS 表格属性.....85
3.4.1 属性值与单位.....52	3.11 页面布局.....86
3.4.2 字体属性.....53	3.11.1 页面布局版式.....86
3.4.3 文本属性.....55	3.11.2 CSS 页面布局技术.....87
3.4.4 背景属性.....56	3.11.3 传统表格布局技术.....91
3.5 样式层叠性.....57	习题.....93
3.5.1 文档结构.....57	<b>第 4 章 JavaScript 编程基础</b> .....100
3.5.2 样式继承.....58	4.1 JavaScript 简介.....100
3.5.3 样式层叠.....58	4.1.1 了解 JavaScript.....100
3.5.4 结构性选择器.....59	4.1.2 JavaScript 应用.....101
3.5.5 伪类.....60	4.1.3 编写 JavaScript 程序的工具.....102
3.5.6 伪元素.....61	4.2 在 HTML 文档中使用 JavaScript.....102
3.6 元素框模型.....62	4.2.1 嵌入 JavaScript 脚本.....102
3.6.1 框模型概述.....62	4.2.2 链接 JavaScript 脚本文件.....104
3.6.2 框属性.....63	4.3 数据类型与常量.....105
3.6.3 外边距重叠.....66	4.3.1 数值型.....105
3.6.4 框大小.....66	4.3.2 字符串.....105
3.7 元素定位.....68	4.3.3 布尔型.....107
3.7.1 定位概念.....68	4.3.4 其他类型的常量.....107
3.7.2 四种定位方式.....69	4.4 变量.....107
3.8 元素布局.....71	4.4.1 变量命名.....107
3.8.1 布局属性.....71	4.4.2 变量声明.....107
3.8.2 浮动与清除.....71	4.4.3 变量赋值.....108
3.8.3 显示和隐藏.....72	4.4.4 变量取值.....108
3.8.4 溢出与剪裁.....73	4.4.5 变量类型.....109
3.8.5 鼠标形状.....74	4.4.6 简述变量的作用域.....109



4.5 运算符与表达式	109	第 6 章 函数	147
4.5.1 基本概念	109	6.1 函数概述	147
4.5.2 运算符的优先级	110	6.1.1 什么是函数	147
4.5.3 运算符的结合性	111	6.1.2 结构化程序设计	148
4.5.4 表达式中的类型转换	111	6.2 使用预定义函数	149
4.6 JavaScript 运算符	112	6.3 函数定义和函数调用	150
4.6.1 算术运算符	112	6.3.1 函数定义	150
4.6.2 关系运算符	113	6.3.2 函数调用	151
4.6.3 逻辑运算符	114	6.4 函数参数的使用	152
4.6.4 位操作运算符	115	6.4.1 给函数添加参数	152
4.6.5 赋值运算符	116	6.4.2 使用多个参数	153
4.6.6 条件运算符	117	6.5 使用函数返回值	154
4.6.7 其他运算符	118	6.5.1 给函数添加返回值	154
习题	119	6.5.2 区分函数和过程	154
<b>第 5 章 基本流程控制</b>	122	6.6 函数的嵌套调用	155
5.1 使用对话框	122	6.6.1 函数嵌套调用的形式	155
5.1.1 警示对话框	122	6.6.2 解决嵌套调用引起的效率问题	157
5.1.2 确认对话框	123	6.7 递归函数	157
5.1.3 提示对话框	123	6.8 变量的作用域	159
5.2 顺序结构	125	6.8.1 全局变量作用域	159
5.3 分支结构	126	6.8.2 使用局部变量避免冲突	160
5.3.1 if 语句	126	6.8.3 全局变量的隐式声明	161
5.3.2 switch 语句	131	习题	162
5.4 循环结构	133	<b>第 7 章 对象编程</b>	164
5.4.1 for 语句	133	7.1 初探对象编程	164
5.4.2 while 语句	135	7.2 对象的基本概念	165
5.4.3 do while 语句	136	7.2.1 什么是对象	165
5.4.4 在循环中使用 break 和 continue 语句	138	7.2.2 对象的属性和方法	165
5.4.5 循环的嵌套	139	7.2.3 类与类的实例	166
5.5 调试程序	141	7.2.4 对象创建与引用	166
5.5.1 排除语法错误	141	7.2.5 JavaScript 对象的分类	168
5.5.2 排除语义错误	142	7.3 使用内置对象	168
5.5.3 使用 IE 8.0 的脚本调试功能	142	7.3.1 Math 对象	168
习题	145	7.3.2 Number 对象	169
		7.3.3 Date 对象	170

7.3.4	String 对象	172	outerText 和 outerHTML 属性	214	
7.3.5	Array 对象	172	8.4 使用样式对象	216	
7.4	自定义对象	176	8.4.1	Style 对象	216
7.4.1	面向对象语言的特征	176	8.4.2	StyleSheet 对象	217
7.4.2	定义属性	176	8.4.3	CurrentStyle 对象	219
7.4.3	定义对象	177	习题	220	
7.4.4	定义方法	179	<b>第 9 章 事件驱动编程</b>	224	
7.4.5	关键字 this	181	9.1 基本概念	224	
7.4.6	定义对象类	182	9.2 事件绑定	225	
7.4.7	继承	186	9.2.1 事件属性及其分类	225	
习题		188	9.2.2 静态绑定	226	
<b>第 8 章 浏览器对象和 HTML DOM</b>		191	9.2.3 动态绑定	227	
8.1	BOM 对象	191	9.2.4 绑定多个事件处理函数	227	
8.1.1	BOM 概述	191	9.2.5	onload 事件	228
8.1.2	窗口 (Window) 对象	192	9.3 使用事件对象	230	
8.1.3	浏览器程序 (Navigator) 对象	195	9.3.1	Event 对象	230
8.1.4	屏幕 (Screen) 对象	196	9.3.2	事件流	231
8.1.5	地址 (Location) 对象	196	9.3.3	阻止事件的默认行为	232
8.1.6	历史 (History) 对象	197	9.4 处理鼠标事件	232	
8.1.7	文档 (Document) 对象	198	9.4.1	鼠标事件	232
8.2	访问 HTML DOM 对象	200	9.4.2	鼠标点击	233
8.2.1	DOM 概念	200	9.4.3	鼠标移动	234
8.2.2	DOM 对象	201	9.4.4	鼠标位置	234
8.2.3	访问集合对象	204	9.5 处理键盘事件	235	
8.2.4	访问指定元素	207	9.5.1	键盘事件	235
8.2.5	访问相关元素	208	9.5.2	识别键盘按键	235
8.3	操纵 HTML DOM 对象	210	9.5.3	识别组合键	236
8.3.1	处理元素的属性	210	9.6 处理表单事件	236	
8.3.2	创建元素	211	9.6.1	访问表单和表单控件	236
8.3.3	删除元素	212	9.6.2	表单提交与验证	239
8.3.4	替换元素	213	9.6.3	处理按钮	240
8.3.5	复制元素	213	9.6.4	处理文本框	241
8.3.6	移动元素	214	9.6.5	处理单选框和复选框	242
8.3.7	使用 innerText、innerHTML、		9.6.6	处理列表框	243
			9.7 处理编辑事件	246	



9.7.1 访问选中区 .....	246	10.1.3 操控滚动区 .....	260
9.7.2 处理复制、剪切和粘贴操作 .....	248	10.1.4 获取浏览器窗口的浏览区 和页面大小 .....	262
9.7.3 处理拖放操作 .....	249	10.2 CSS 滤镜 .....	262
9.8 处理异常 .....	251	10.3 文字特效 .....	265
9.8.1 运行时错误 .....	251	10.4 图片特效 .....	268
9.8.2 onerror 事件 .....	252	10.5 时间特效 .....	269
9.8.3 try...catch 语句 .....	253	10.6 窗体特效 .....	272
习题 .....	254	10.7 鼠标 .....	276
<b>第 10 章 JavaScript 网页特效 .....</b>	<b>257</b>	10.8 菜单 .....	278
10.1 操控元素的大小与位置 .....	257	习题 .....	284
10.1.1 设置元素的大小与位置 .....	257	参考文献 .....	285
10.1.2 获取元素的大小与位置 .....	258		

# 第 1 章

## Web 技术概述

本章简要介绍 Web 的基础知识、技术和标准。目前，Web 标准化已得到 Web 业界的广泛认同，成为人们选用 Web 开发技术、工具和浏览器的准则之一。

### 1.1 Internet 简述

#### 1.1.1 Internet 定义

Internet（中文译名“因特网”，常称为“国际互联网”或“互联网”）是一种以 TCP/IP 连接全球各种计算机的广域网，具有图 1.1 所示的“网际网”结构。其中，与 Internet 相连的任何一台计算机都称为主机。

Internet 提供的基本服务有全球信息网（Web）、电子邮件（E-mail）、文件传输（FTP）、远程登录（Telnet）、网络新闻（NetNews）、电子公告栏（BBS）等服务。

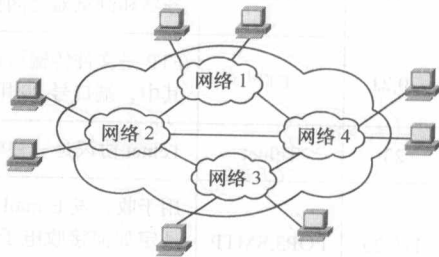


图 1.1 Internet 的“网际网”结构

#### 1.1.2 IP 地址与域名

##### 1. IP 地址

IP 地址是标识 Internet 每台主机的唯一地址，具有固定、规范的格式。IP 地址由 4 个字节组成，书写为带圆点的十进制数形式，即每个字节用一个十进制数（0~255）表示，而各个字节之间由圆点“.”分隔，如 202.108.9.16 就是一个标准的 IP 地址。

##### 2. 域名

因 IP 地址不易记忆，故引入域名机制。通过为每台主机建立 IP 地址与域名之间的映射关系，用户可以直接使用域名来访问主机。一台主机的域名由用圆点“.”隔开的多级域名序列组成，通常具有如下格式：

四级域名.三级域名.二级域名.顶级域名

例如，域名“www.sysu.edu.cn”表示的主机是隶属中国（cn）教育机构（edu）的中山大学（sysu）的提供 Web 服务的主机。

### 1.1.3 TCP/IP

TCP/IP 是针对 Internet 开发的一种网络协议标准,其目的在于解决异种计算机网络的通信问题。随着 Internet 的广泛应用, TCP/IP 已成为最常用的网络协议。

TCP/IP 主要包括以下两个子协议。

(1) IP (Internet Protocol, 即互联网协议): 其基本任务是在 Internet 中传送 IP 数据包, 包括数据包的传送、路由选择和拥塞控制等功能。IP 数据包不仅包含数据, 也包含发送主机和接收主机的 IP 地址。

(2) TCP (Transmission Control Protocol, 即传输控制协议): 是一种可靠的、基于连接的传输层协议, 保证信息能够无差错地传输到目的主机上的应用程序。TCP 包括差错控制、数据包排序和流量控制等功能。

为了区分同一台主机上不同的 Internet 应用程序, TCP 在数据包中增加一个称为端口号的数值 (在 0 ~ 65 535 之间)。其中, Internet 的标准服务 (如表 1.1 所示) 使用小于 1 024 的端口号。例如, 80 表示实现 Web 服务的 HTTP 任务, 而 21 表示 FTP 任务。

表 1.1 常用的 TCP 端口号

端口号	协议	说明
80	HTTP	HTTP 是超文本传输协议 (HyperText Transfer Protocol) 的简称, 规定如何在 Web 服务器和浏览器之间传输消息
20,21	FTP	FTP 是文件传输协议 (File Transfer Protocol) 的简称, 规定如何在网络上传送文件。其中, 端口号 20 用于传输数据, 而 21 用于传输控制信息
23	Telnet	Telnet 协议是一种模拟终端协议, 允许将本地计算机连接到远程主机
110,25	POP3,SMTP	用于收、发 E-mail。其中, POP3 是邮局协议第 3 版 (Post Office Protocol 3) 的简称, 规定如何接收电子邮件, 使用端口 110; SMTP 是简单邮件传输协议 (Simple Mail Transfer Protocol) 的简称, 规定如何发送和中转电子邮件, 使用端口 25
161,162	SNMP	SNMP 是简单网络管理协议 (Simple Network Management Protocol) 的简称, 规定如何在网络设备之间相互传送管理信息
≥1024	自定义	供用户自己定义端口号的用途, 没有通用性

## 1.2 Web 基本知识

### 1.2.1 Web 定义

WWW 是 World Wide Web (全球广域网) 的简称, 常称为 Web (万维网)。它是以 HTML 语言和 HTTP 为基础、提供面向 Internet 服务、支持一致用户界面的全球信息网络, 具有交互性、动态性和多平台等特性。

在 Web 中, 主要包括以下两类主机。

- (1) Web 客户机：安装有 Web 浏览器（如 IE）的计算机。
- (2) Web 服务器：提供 Web 信息服务的计算机，一般是高性能服务器。

如图 1.2 所示，对 Web 的访问过程大致分为以下 4 步。



图 1.2 Web 访问流程

- (1) 用户在 Web 客户机上输入网址发出浏览网页的请求（HTTP 请求）。
- (2) 该请求通过 Internet 从 Web 客户机传输到 Web 服务器。
- (3) Web 服务器接收这个请求，据此找到或生成相关网页，然后发送给 Web 客户机。
- (4) 请求的网页又通过 Internet 传输到 Web 客户机，然后由 Web 浏览器显示。

## 1.2.2 HTTP

HTTP 是 Web 浏览器和服务器用来交换信息的一种 Internet 应用协议，该协议允许用户使用一个客户端程序（如 Web 浏览器）通过 URL 在 Web 服务器上检索文本、图像、声音等信息。

HTTP 具有以下两个显著特性。

- (1) 无连接性：是指 Web 服务器对客户机的每次连接只处理一个请求。当服务器每次接收到客户机的请求时，就会建立一个新连接，处理完成后立即断开这次连接。
- (2) 无状态性：HTTP 不保存事务或状态，后续事务所需的状态信息必须在协议之外完成。

## 1.2.3 HTML 文档

### 1. 超文本

超文本（Hypertext）也是普通的文本，但是可以通过标签控制文本的显示格式，也可以嵌入链接、描述声音、图像、视频、动画等多媒体信息，因此超文本也常称为超媒体（Hypermedia）。

### 2. 超链接

超链接（Hyperlink）是指从一个网页指向另一个目的端的链接，这个目的端通常是另一个网页，也可以是电子邮件地址、其他文档文件（如多媒体文件、Word 文档等）、程序或者当前网页中的其他位置。在浏览器中，超链接通常显示为带下画线的彩色字符串或图片。

### 3. HTML 语言

HTML 是超文本标记语言（HyperText Markup Language）的简称，用于描述网页。HTML 通过在正文文本中嵌入各种标签，使普通正文文本具有超文本的功能。

### 4. HTML 文档

HTML 文档常称为网页（WebPage）、页面，是符合 HTML 语言规则的超文本文件，其最常见扩展名是“.htm”和“.html”。

网页是 Web 信息的基本单元，存放在各个 Web 服务器（或称 Web 站点）上。

## 1.2.4 网站

网站 (Web Site) 是多个网页的集合, 按网站内容可将网站分为以下 4 类。

(1) 门户网站: 是一种综合性网站, 涉及领域非常广泛, 包含文学、音乐、影视、体育、新闻和娱乐等多个方面的内容。例如, 网易 (www.163.com)。

(2) 个人网站: 是以个人名义创建的网站, 其内容、样式、风格等都具有较强个性化。

(3) 专业网站: 是具有很强专业性的网站, 通常只涉及某一个领域, 内容专业。例如, 太平洋电脑网 (www.pconline.com.cn)。

(4) 职能网站: 是具有专门功能 (如政府职能、电子商务、搜索引擎等) 的网站, 如阿里巴巴 (china.alibaba.com)、当当网上商城 (home.dangdang.com) 等。

访问网站时, 把只使用域名 (如 http://www.sysu.edu.cn) 就可浏览到的第 1 个页面称为该网站的主页 (Home Page) 或首页。通过主页中的超链接, 可以浏览该网站的内部页面 (称为内页)。

## 1.2.5 网址

### 1. URL

URL 是统一资源定位符 (Uniform Resource Locator) 的简称, 用于唯一确定 Web 资源 (如网页、图像、音频、程序等) 的位置。URL 也常称为 Web 地址或网址, 其一般格式如下。

Protocol://Host:Port/Path

例如, http://www.sysu.edu.cn/2003/xxgk/xxgk.htm 就是一个典型的 URL 网址。

(1) Protocol: 协议名, 如 “http://”、“ftp://” 等。当该部分省略时, 缺省是 “http://”。

(2) Host: 存放资源的主机 IP 地址或域名。

(3) Port: 端口号, 通常不写。其默认值依赖于特定协议, 如 HTTP 默认使用端口 80。

(4) Path: 指定资源在服务器中的位置, 类似文件路径。

URL 是目前最常用的资源标识格式, 但存在缺陷, 即由于 URL 强调定位, 故当改变资源的存放位置时, 也必须相应改变其 URL。

### 2. URN

URN 是统一资源名 (Uniform Resource Name) 的简称, 用于为 Web 资源给出一个唯一名称。与 URL 不同, URN 与地址无关。由于 URN 不依赖于资源的存放位置, 能够弥补 URL 的缺点, 因此 URN 将逐渐流行。

### 3. URI

URI 是统一资源标识符 (Uniform Resource Identifier) 的简称, 用于唯一标识 Web 中的资源。URI 是 URL 和 URN 的统称, 即任何一个 URL、URN 都称为 URI 标识符。

# 1.3 Web 浏览器与 Web 服务器

## 1.3.1 Web 浏览器

### 1. Web 浏览器定义



Web 浏览器 (Browser) 是一种显示网页、并允许用户与网页互动的 Web 客户端程序, 这些网页既可以来自 Web 服务器, 也可以是本地网页文件。

目前,常用的浏览器有 IE、Firefox、Opera、Safari、Maxthon 和 Mosaic<sup>1</sup>等。使用这些浏览器的最基本方法是输入网址、单击超链接以及单击工具栏中的“后退”或“前进”按钮。

## 2. IE 浏览器

Windows Internet Explorer (简称 IE) 是微软推出的、目前使用最广泛的 Web 浏览器,其最大特点是 IE 捆绑于 Windows 操作系统,易于 Windows 用户使用。IE 的最早正式版 1.0 发布于 1995 年 8 月,至 2009 年 3 月微软已推出最新版本 IE8.0。

由于 IE8.0 之前的版本 IE6.0 (2001 年 8 月) 和 IE7.0 (2006 年 11 月) 对 Web 标准支持不完善,特别是 IE6.0 安全性较差、易受网络攻击,因此微软强烈建议用户使用 IE8.0 或其后续版本。

从外观上来看,IE8.0 与 IE6.0 的最大区别是 IE8.0 默认采用选项卡式浏览,从而可以在单个浏览器窗口中同时打开多个网页。如图 1.3 所示,在“选项卡”栏中,单击“新选项卡”按钮  可以打开新的空白选项卡,单击选项卡可切换当前选项卡,而单击当前选项卡中的“关闭”按钮  则可关闭选项卡。操作时,注意地址栏中的网址将与当前选项卡保持一致。

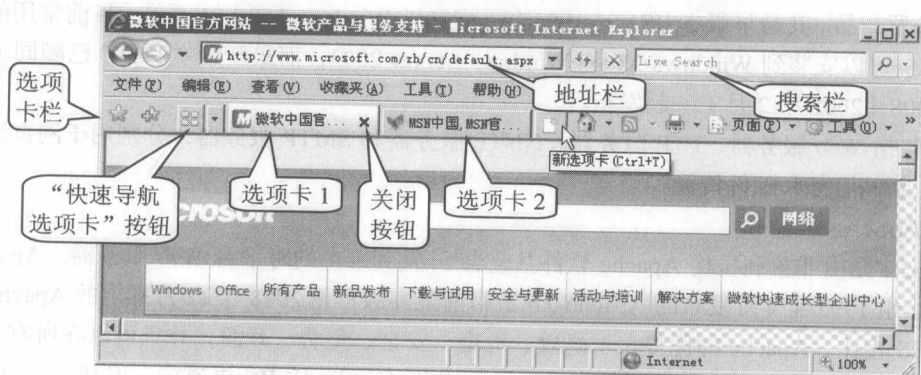


图 1.3 在 IE8.0 中使用多个选项卡同时打开多个页面

用户也可以改变 IE8.0 的选项卡操作特性,方法是:通过执行菜单“工具”→“Internet 选项”命令打开“Internet 选项”对话框,然后单击其“常规”选项卡中“选项卡”部分的“设置”按钮,打开“选项卡浏览设置”对话框。此时,若清除“启用选项卡式浏览”复选框,则将关闭选项卡式浏览,从而使 IE8.0 的操作外观与 IE6.0 类似。

## 3. Firefox 浏览器

Mozilla Firefox (常称为 Firefox,火狐)是由 Mozilla 基金会<sup>2</sup>与开源团体共同开发的 Web 浏览器。其第一个正式版 Firefox1.0 发布于 2004 年 11 月,目前常用的 Firefox3.X 始于 2008 年 6 月。近年来,Firefox 发展迅猛,已成为普及度仅次于 IE 的浏览器。

与 IE 相比,Firefox 具有跨平台(支持 Windows、Linux、MacOSX、OS/2 等操作系统)、支持的 Web 标准更全面和安全性更强等特色。

目前,Firefox 和 IE (7.0 与 8.0) 的操作界面和功能已非常相似,主要不同只是有关术语而已。对于熟悉 IE 的读者,可以在 Firefox 中使用菜单“帮助”→“致 Internet Explorer 用户”命令打开

<sup>1</sup> 1993 年,美国伊利诺斯大学的 NCSA 组织发布 Mosaic 浏览器,使之成为 Web 发展初期最具代表性的浏览器。

<sup>2</sup> Mozilla 基金会是为支持和领导开源的 Mozilla 项目而于 2003 年 7 月设立的一个非营利组织,其主要项目是 Firefox 浏览器和 Thunderbird 电子邮件客户端程序。Mozilla 基金会源于网景公司,而网景公司推出了 20 世纪 90 年代非常流行的 Netscape Navigator (导航者) 浏览器。



“致 IE 用户”页面,从中查知有关术语在这两种浏览器中的差异,如 IE 术语“选项卡式浏览”等同於 Firefox 术语“标签式浏览”。

## 1.3.2 Web 服务器程序

### 1. Web 服务器程序定义

Web 服务器程序是指在 Web 服务器主机上运行的、提供 Web 信息服务的程序,常简称为 Web 服务器。由于 Web 服务器使用 HTTP 与 Web 浏览器进行信息交流,因此 Web 服务器也称为 HTTP 服务器。

目前,最著名的 Web 服务器是微软的 IIS 和免费的 Apache。它们都支持 ASP、PHP 和 JSP 等主流动态网站技术,但相对来说,Apache 支持的开发语言更多,而 IIS 则偏重于支持 ASP。

### 2. IIS

IIS (Internet Information Server, 互联网信息服务器)是微软推出的、基于 Windows 系统的 Web 服务器产品。其最早版本 IIS1.0 (1995 年)配置于 Windows NT 3.51 系统,目前常用的 IIS 6.0 (2003 年)可以安装到 Windows XP 和 Windows Server 2003,而最新版本 IIS7.0 已随同 Windows Server 2008 (2008 年 2 月)一起发布。

IIS 包括 Web 服务器、FTP 服务器、NNTP 服务器和 SMTP 服务器,分别用于网页浏览、文件传输、新闻服务和邮件传输。

### 3. Apache

Apache 是由非盈利机构 Apache 软件基金会<sup>1</sup>开发和维护的跨平台 Web 服务器。Apache 始于 Apache 小组 1995 年 4 月发布的 Apache 0.6.2,目前最高版本是 2008 年 12 月推出的 Apache 2.2.11。

与 IIS 相比,Apache 的优势在于高效、稳定、安全、免费、开源,并且可以在所有主流操作系统(如 Windows、Unix、Linux 等)上运行,因此 Apache 比 IIS 更流行;但缺点是 Apache 没有为管理员提供图形操作界面,较难使用。

## 1.4 动态网页及相关技术

动态网页有两种含义,一种是客户端动态网页,另一种是服务器端动态网页。

### 1.4.1 客户端动态网页

客户端动态网页称为动态 HTML (Dynamic HTML, 简称 DHTML),是一种即使在网页下载到浏览器以后仍然能够随时变换的网页。例如,当鼠标移至文章段落中,段落能够变成蓝色,或者网页标题能够滑过电脑屏幕,等等。反之,没有动态效果的网页称为(客户端)静态网页。

DHTML 有 3 个主要特征,即动态样式、动态内容和动态定位。动态样式是指改变网页的外部显示特征,动态内容是指更换显示在页面上的文本或图像,而动态定位是指移动页面上的文本、图像等页面元素。

DHTML 是一种通过各种技术的综合发展而得以实现的概念,这些技术包括脚本语言(如常

<sup>1</sup> Apache 软件基金会(即 Apache Software Foundation, 简称 ASF)创建于 1999 年,是专门为支持 Apache 开源项目而建立的非盈利性组织,其主要成果有 Apache HTTP Server、Ant、iBATIS、Jakarta、Logging、Maven、Struts、Tomcat 和 Tapestry 等。

用的 JavaScript、VBScript)、DOM 和 CSS 等。

## 1.4.2 服务器端动态网页

基于 Web 服务器角度, 动态网页是指采用动态网站技术实时动态生成的网页, 而其他直接发送给浏览器的网页则称为静态网页。动态网站是指支持网页动态生成的网站, 否则称为静态网站。

目前, 常用的动态网站技术有 ASP、PHP、JSP 等。这几种技术都具有在 HTML 代码中混合某种程序代码、并由脚本语言引擎解释执行的能力。

### 1. ASP

目前, ASP (Active Server Pages, 活动服务器页面) 分为传统 ASP 和 ASP.NET, 但术语 ASP 通常是指传统 ASP。

传统 ASP 是微软 1996 年 11 月推出的服务器端脚本编写和运行环境, 使用它可以创建和运行动态页面和 Web 应用程序。ASP 页面主要运行于 IIS, 可以嵌入 VBScript、JScript 等脚本程序, 其文件扩展名是 .asp。

ASP.NET 是微软在 2001 年推出的基于 .NET 框架的新型动态网站开发技术。与传统 ASP 相比, ASP.NET 具有编译执行、代码分离 (即将 HTML 代码与服务器端程序分离) 和支持新型编程语言 (即支持 C#、J# 和 VB.NET, 放弃支持 VBScript) 等特点, 并且 ASP.NET 页面文件扩展名是 .aspx。

尽管 ASP.NET 功能强大, 但对于初学者而言, 传统 ASP 仍然是学习动态网站开发技术的简捷工具。

### 2. PHP

PHP (Hypertext Preprocessor, 超文本预处理器) 是一种跨平台的服务器端 HTML 嵌入式脚本语言, 其语法类似 C、Java 和 Perl 语言。PHP 具有开源、免费、简单、易扩展等特点, 可用于编写 PHP 动态网页。PHP 页面文件的默认扩展名是 .php。

PHP 最早由 Rasmus Lerdorf 于 1995 年 6 月推出, 目前流行的是始自 2004 年 7 月发布的 PHP5.X。

### 3. JSP

JSP (Java Server Pages, Java 服务器页面) 是由 Sun Microsystems 公司倡导、许多公司参与一起建立的基于 Java Servlet 和 Java 体系的服务器端动态网页技术标准, 具有简单易用、完全面向对象、跨平台和安全可靠等特点。与 ASP 类似, JSP 在 HTML 中嵌入 Java 程序段, 从而形成 JSP 动态网页文件 (\*.jsp)。

JSP 的最早正式规范 JSP1.0 于 1999 年 9 月推出, 目前流行的是始自 2003 年发布的 JSP2.X 规范。

## 1.5 Web 标准化

### 1.5.1 Web 标准

Web 标准是指由 W3C<sup>1</sup>、ECMA<sup>2</sup>等标准化组织制订的一系列 Web 技术规范总称, 主要包括

<sup>1</sup> W3C 是万维网组织 (World Wide Web Consortium) 的缩写, 由 Web 发明者 Tim Berners-Lee 领导, 成立于 1994 年, 网址是 <http://www.w3.org/>。其主要工作是研究和制定开放的 Web 规范, 以便提高 Web 相关产品的互用性。

<sup>2</sup> ECMA 是欧洲计算机制造协会 (European Computer Manufacturer's Association) 的简称, 创建于 1961 年, 网址是 <http://www.ecma-international.org/>。其主要工作是致力于信息和通信领域的标准化。

结构标准、表现标准和行为标准 3 部分 (如表 1.2 所示), 分别用于规范网页的内容结构、表现格式和动态行为。

表 1.2

目前常用的 Web 标准

类别	标 准	说 明
结构标准	HTML	HTML 是网页的基本描述语言, 常用版本是 HTML4.0 (1998 年 4 月)
	XML	XML 是可扩展标记语言 (eXtensible Markup Language) 的缩写, 常用版本是 XML1.0 (1998 年 2 月)
	XHTML	XHTML 是可扩展超文本标记语言 (eXtensible HyperText Markup Language) 的缩写, 是符合 XML 规范的 HTML。其常用版本是等同于 HTML4.01 的 XHTML1.0 (2000 年 1 月)
表现标准	CSS	CSS 是层叠样式表 (Cascading Style Sheets) 的缩写, 目前版本有 CSS 1.0 (1996 年 12 月)、CSS 2.0 (1998 年 5 月) 和 CSS 3.0 (2004 年 4 月)。其中, CSS 2.1 最常用
行为标准	DOM	DOM 是文档对象模型 (Document Object Model) 的缩写, 目前版本有 DOM1.0 (1998 年 10 月)、DOM2.0 (2000 年 11 月) 和 DOM3.0 (2004 年 4 月)
	ECMAScript	ECMAScript 是 ECMA 制定的标准脚本语言, 目前常用版本是 ECMAScriptv3 (1999 年 12 月)。而 JavaScript 和 JScript 是对 ECMAScript 标准的实现和扩展, 这几个术语常统称为 JavaScript

要了解这些规范的详细说明, 读者可以访问 <http://www.w3.org>, 或者使用搜索工具 (如 <http://www.baidu.com>) 按“XHTML 手册”、“CSS 手册”、“JavaScript 手册”或“DOM 手册”等关键字搜索下载由 Web 技术专业人士整理的参考手册。

此外, 通过访问 <http://validator.w3.org> 网站, 读者可以使用 W3C 提供的工具验证所编写的网页是否符合 Web 规范。

## 1.5.2 制作符合 Web 标准的网页

从网页制作的角度来看, Web 标准化是指在制作网页时遵循 Web 规范并采用相关理念。目前, 制作标准化网页的一般方法如下。

(1) 采用 XHTML+ CSS + JavaScript 技术, 以实现网页结构、表现和行为的分离。即明确区分定义网页结构、表现和行为的 HTML、CSS 和 JavaScript 代码, 甚至放入不同的文件。

(2) 正确使用 HTML 标签, 即编写结构化的、有语义的 HTML。例如, 网页的主标题使用<h1>标签, 子标题则使用<h2>至<h6>之一, 段落用<p>标签, 强调的文字应当用<em>而不是<i>, 等等。此外, 合理使用<div>、<span>标签及相关 id 和 class 属性, 从而有语义地扩展 HTML 的块级标签和行内标签。

(3) 使用 CSS 处理网页的外观。

(4) 依靠 JavaScript 去增强而不是替代网站的特征。例如, 没有必要使用 JavaScript 去实现那些易于只由 CSS 实现的特殊效果。

制作符合 Web 标准的网页及网站有如下好处。

(1) 对于网站浏览者而言, 能够提高网站的易用性。不仅能够加快网页显示速度, 也能够使网站被更多的用户 (包括失明、视弱、色盲等残障人士) 与设备 (包括屏幕阅读机、手持设备、搜索机器人、打印机、电冰箱等) 所访问。