

100%

宝典丛书

100万

内容丰富、权威

提供最新、最全的文档对象模型参考和JavaScript核心语言参考

通过由浅入深、由易到难的典型示例程序，快速精通JavaScript脚本编程

利用JavaScript快速创建具有交互性的动态网页

JavaScript

编程

宝典

袁建洲 尹喆 等编著

可以访问

“华信卓越”公司

网站 (www.hxex.cn) 的

“资源下载”栏目查找并

下载书中示例程序的源代码



电子工业出版社

Publishing House of Electronics Industry
<http://www.phei.com.cn>

内 容 简 介

JavaScript 脚本用于创建 Web 浏览器支持的交互式网页，是一种简单易学的 Web 编程语言。本书全面系统地介绍了客户端 JavaScript 脚本编程技术。从简单介绍 JavaScript 与 Web 开始，循序渐进地介绍了 Web 浏览器、基本的编程技巧和编程步骤；接着，深入探讨了浏览器文档对象模型的发展，详细介绍了 JavaScript 对象的有关知识，包括对象的属性、方法和事件处理程序等内容；最后，细致地讲述了 JavaScript 核心语言，包括 String, Math, Date 和 Array 等核心对象以及异常处理等内容。

本书内容全面、示例典型，适合各层次的网页设计人员学习和参考。通过本书的学习，读者可以根据实际需要制作出自己的动态网页，全方位拓展自己的网页设计知识和基本技能。

**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

JavaScript 编程宝典 / 袁建洲, 尹喆等编著. —北京: 电子工业出版社, 2006.1

(宝典丛书)

ISBN 7-121-02156-0

I .J... II .①袁...②尹... III .JAVA 语言 - 程序设计 IV .TP312

中国版本图书馆 CIP 数据核字 (2005) 第 152376 号

责任编辑: 牛晓丽 张月萍 梁卫红

排版制作: 华信卓越公司制作部

印 刷: 北京东光印刷厂

出版发行: 电子工业出版社

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: 59.25 字数: 1687 千字

印 次: 2006 年 1 月第 1 次印刷

定 价: 98.00 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) 68279077。质量投诉请发邮件到 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

随着 Internet/Intranet 技术的发展, Web 应用已成为其最重要的部分之一。以前, HTML 文档只能生成静态的网页, 不能产生交互和动态效果, JavaScript 脚本语言正是在这种情况下应运而生的。

JavaScript 脚本是一种在客户端运行的解释性语言, 没有集成开发环境, 使用文本编辑器即可完成 JavaScript 脚本的编辑和修改。将包含 JavaScript 脚本的 HTML 文档载入到支持 JavaScript 脚本的 Web 浏览器中, JavaScript 脚本代码就可以被解释并执行。因此, 使用 JavaScript 进行脚本编程只需在计算机上安装文本编辑器和支持 JavaScript 脚本的 Web 浏览器即可。

本书的组织结构

本书分 5 个部分来由浅入深地介绍 JavaScript 编程技术。

第 1 部分 JavaScript 入门

第 1 部分的第 1 章简单介绍了 Web 技术以及 JavaScript 脚本语言的特点和主要应用场合。由于 JavaScript 脚本是嵌入在 HTML 文档中的编程语言, 因此, 第 2 章介绍了 HTML 文档基础和 HTML 常用标记。第 3 章介绍了 JavaScript 的编程环境、编程步骤和编程策略, 并指导读者编写出第一个脚本。

第 2 部分 JavaScript 脚本指南

第 2 部分是针对 JavaScript 初学者编写的, 首先介绍了 JavaScript 脚本的语言基础、基本语句以及基于对象的编程, 然后介绍了如何利用 JavaScript 核心对象和文档对象模型中的对象编写一些简单的脚本网页。本部分的目的是使读者逐步了解 JavaScript 编程知识并掌握编程技巧, 这有助于理解后面章节介绍的内容和编程示例。

第 3 部分 文档对象模型参考

第 3 部分是内容最多的一部分, 它提供了对于新旧浏览器中文档对象模型的深入探讨。在大多数的 JavaScript 脚本编程中, 都需要使用文档对象模型生成对象的属性、方法和事件处理程序来实现网页的动态和交互效果。对于 JavaScript 编程来说, 理解文档对象模型以及元素对象的属性、方法和事件处理程序非常重要。由于文档对象模型在 Web 浏览器发展过程中是不统一的, 因此, 在介绍元素对象以及属性、方法和事件处理程序时, 提供 Web 浏览器版本支持的兼容性列表。

第 4 部分 JavaScript 核心语言参考

第 4 部分中包括了 JavaScript 核心语言的参考内容。作为一种脚本编程语言, JavaScript 具有自己的核心语言功能, 这些核心语言功能独立于文档对象模型, 提供丰富的编程功能。

与第 3 部分一样, 在介绍 JavaScript 核心语言时, 也提供 Web 浏览器版本支持的兼容性列表。

第 5 部分 附录

附录 A 提供了 JavaScript 保留字列表。书中示例的源代码放在网上，可以访问“华信卓越”公司网站 (www.hxex.cn) 的“资源下载”栏目查找并下载，附录 B 对此进行了说明。要运行这些脚本代码，最好在计算机上安装 Internet Explorer 6 (或更高版本) 和 Netscape Navigator 7 (或更高版本)。

学习 JavaScript 的必备知识

本书虽然不要求读者具有丰富的编程经验，但是如果使用 HTML 创建过一些网页，就比较容易理解 JavaScript 是如何与网页上的常用元素进行交互的。有时，需要修改 HTML 标记语言来编写脚本。如果已经很熟悉那些标记，就可以很容易地融会贯通 JavaScript 的增强功能。

表单及其元素在许多典型的 JavaScript 程序中扮演着十分重要的角色。读者应该熟悉这些元素和它们的 HTML 属性。不过，读者不必了解服务器脚本以及如何将信息从表单传递到服务器。本书内容主要集中在客户端脚本上，在浏览器完全载入 JavaScript 增强的 HTML 网页后，这些脚本的运行与服务器无关。

另外，读者应熟悉最新文档标记标准，如 XML 和层叠样式表等。

本书约定

本书的脚本清单使用等宽字体表示，这样可使它们与正文的其他部分区分开来。由于页面宽度的限制，脚本清单可能常常断行。如果在文档中输入脚本清单时，这些断行导致代码运行错误，那么可在“华信卓越”公司网站 (www.hxex.cn) 的“资源下载”栏目中找到相应的源代码。

在第 3 部分和第 4 部分的兼容性列表中，指定了对象模型和语言特性在各种浏览器文档中的支持情况。为了方便，用首字母缩写和版本号来表示浏览器名称。例如，WinIE5 表示运行于 Windows 操作系统上的 Internet Explorer 5，NN6 表示运行于任何操作系统上的 Netscape Navigator 6，Moz 表示基于 Mozilla 的相对较新的浏览器，Safari 表示运行在 Mac OS X 操作系统上的 Apple 自己的浏览器。如果某个功能随着浏览器的某一版本引入且后续版本都对其提供支持，那么版本号后面将有一个加号 (+)。例如，标记为 WinIE5.5+ 的功能表示该功能至少需要运行在 Windows 操作系统上的 Internet Explorer 5.5 中，同时，WinIE6 及未来的 WinIE 版本也支持该功能。有时，一个功能或行为只适用于一个操作系统，例如，标记为 NN4 的功能意味着只能运行于 Netscape Navigator 4.x 中。减号 (例如 WinIE-) 表示此浏览器不支持该功能。

为了强调最新 Web 标准，本书的 HTML 和代码清单遵循 XHTML 代码规范，所有的标记及其属性都采用小写字母形式，单标记采用自封闭标记形式 (例如，用 `
` 代替 `
`)。

图标

本书中经常出现以下几个图标，它们分别表示不同的含义：



提供相关内容的补充信息，指出需要注意的事项。



指出对实际编程非常有帮助的线索和技巧。



给出警告，提醒读者应该特别关注正在讨论的内容。



指示在其他章节有与本主题相关的内容。

编 者 序

目前,使网页具有交互能力的脚本编程语言主要分为服务器脚本语言和客户端脚本语言两大类。使用服务器脚本语言编写的交互式代码由Web服务器解释并执行,典型代表是JSP,PHP和ASP等;使用客户端脚本语言编写的交互式代码由Web浏览器解释并执行,典型代表是JavaScript和VBScript。VBScript脚本只适用于Internet Explorer浏览器,而JavaScript脚本可用于几乎所有的Web浏览器,因此,对于客户端脚本编程来说,JavaScript是最流行也是最恰当的选择。

JavaScript在1995年由Netscape提出,当时的名称为LiveScript,很快在Sun公司的支持下更名为JavaScript,并在Netscape Navigator 3.0中开始支持。随后,JavaScript语言规范不断发展,JavaScript数据表现和控制的能力不断完善。目前,JavaScript正在酝酿着问世以来最大规模的改进,引进了类和接口等面向对象语言才具有的语法,其目的是使JavaScript成为功能更加强大的脚本编程语言。

利用JavaScript脚本,可以在客户端直接对网页元素进行编程控制,根据网页需要创建窗口、创建对象、执行数学计算并进行交互式用户操作等,从而产生各种交互和动态效果。

本书主要具有下列特点:

- ◆ 支持最新技术。本书介绍的JavaScript脚本语言支持最新版本的Web浏览器,删除了一些过时的技术。
- ◆ 内容极其丰富。本书介绍的JavaScript脚本编程技术囊括了JavaScript相关技术的各个方面,从JavaScript基本语法到对象参考,从DHTML到CSS,从DOM到XML,都有全面深入的探讨。
- ◆ 提供所有示例代码,而且提供JavaScript脚本在交互数据、交互式用户界面、多框架交互、少量数据查找、动态HTML等7个方面的典型应用代码。

在学习本书的过程中,建议读者多上机进行操作,通过运行和修改这些JavaScript示例代码,可加深对JavaScript脚本编程技术的理解和掌握。

参加本书编写工作的人员有袁建洲、尹喆、胡宝成、饶贵安、夏功成、朱振平、宗利、向双成、于红、朱丽娜、陶国强、徐亚莉、朱芳芳等。

由于编者水平有限,JavaScript脚本涉及的内容又较多,疏漏和不当之处在所难免,恳请广大读者原谅并提出宝贵意见。

编者

2005年12月

目 录

第 1 部分 JavaScript 入门	1
第 1 章 JavaScript 与 Web 基础	2
1.1 Web 技术概述	2
1.1.1 Web 基本结构	2
1.1.2 Web 运行机制	4
1.1.3 辅助程序和插件程序	4
1.1.4 CGI 技术	5
1.1.5 脚本编程	5
1.1.6 Java applet	6
1.2 JavaScript 概述	7
1.2.1 JavaScript 语言简史	7
1.2.2 JavaScript 的主要特点	8
1.2.3 JavaScript 脚本的应用场合	9
第 2 章 HTML 文档	16
2.1 HTML 文档基础	16
2.1.1 HTML 标记	16
2.1.2 HTML 文档的基本结构	17
2.1.3 使用<body>标记设置文档属性	18
2.2 HTML 常用标记	18
2.2.1 文本标记	19
2.2.2 图像标记	28
2.2.3 超链接标记	29
2.2.4 表格标记	30
2.2.5 框架标记	33
第 3 章 JavaScript 编程初步	36
3.1 设置编程环境	36
3.1.1 选择文本编辑器	36
3.1.2 选择浏览器	37
3.2 JavaScript 编程步骤	37
3.2.1 <script>标记	37
3.2.2 <script>标记对位置	38
3.2.3 输入 JavaScript 脚本	40
3.2.4 运行 JavaScript 脚本	41
3.2.5 检查和修改 JavaScript 脚本	41
3.3 JavaScript 编程策略	43
3.3.1 浏览器对 JavaScript 脚本的支持	43
3.3.2 确定用户使用的浏览器类型	44

3.3.3	处理不支持 JavaScript 脚本的情况	45
第 2 部分	JavaScript 脚本指南	49
第 4 章	JavaScript 语言基础	50
4.1	数据	50
4.1.1	基本数据类型	50
4.1.2	常量	51
4.1.3	变量	51
4.1.4	转义字符	52
4.2	操作符	52
4.2.1	算术操作符	52
4.2.2	比较操作符	54
4.2.3	字符操作符	55
4.2.4	逻辑操作符	56
4.2.5	位操作符	57
4.2.6	赋值操作符	59
4.3	表达式	60
4.3.1	操作符优先级	61
4.3.2	数据类型转换	62
第 5 章	JavaScript 基本语句	66
5.1	基本处理流程	66
5.2	分支结构语句	67
5.2.1	if 语句	67
5.2.2	if...else 语句	69
5.2.3	switch 语句	70
5.2.4	?:操作符	73
5.3	循环结构	74
5.3.1	for 语句	74
5.3.2	while 语句	76
5.3.3	do...while 语句	78
5.3.4	break 语句和 continue 语句	79
5.4	函数	80
5.4.1	函数的定义和调用	80
5.4.2	变量作用域	81
5.4.3	JavaScript 全局函数	84
第 6 章	面向对象的编程和基于对象的编程	86
6.1	面向对象的编程	86
6.1.1	对象	86
6.1.2	属性、方法和事件	87
6.1.3	面向对象编程的特点	87
6.1.4	对象组织的层次性	88
6.2	基于对象的编程	88
6.2.1	HTML 结构和 DOM	89

6.2.2	载入文档时	90
6.2.3	对象引用	92
6.2.4	如何定义对象	93
6.2.5	JavaScript 核心对象	96
6.2.6	自定义对象	97
第 7 章	String, Math, Date 和 Array 对象	100
7.1	核心语言对象	100
7.2	String 对象	100
7.2.1	连接字符串	101
7.2.2	字符串方法	101
7.3	Math 对象	103
7.4	Date 对象	104
7.4.1	使用 Date 对象	104
7.4.2	日期计算	105
7.5	Array 对象	106
7.5.1	创建数组	107
7.5.2	存取数组数据	107
7.5.3	关联数组	108
7.5.4	数组中的 document 对象	109
第 8 章	window 对象和 document 对象	110
8.1	顶级对象	110
8.2	window 对象	110
8.2.1	访问 window 对象的属性和方法	111
8.2.2	创建窗口	111
8.3	窗口属性和方法	112
8.3.1	window.status 属性	112
8.3.2	window.alert()方法	113
8.3.3	window.confirm()方法	113
8.3.4	window.prompt()方法	114
8.3.5	onload 事件处理程序	115
8.4	location 对象	115
8.5	navigator 对象	116
8.6	history 对象	116
8.7	document 对象	116
8.7.1	document.forms[]属性	117
8.7.2	document.images[]属性	117
8.7.3	document.title 属性	118
8.7.4	document.write 方法	118
8.7.5	document.createElement()和 document.createTextNode()方法	120
第 9 章	表单和表单对象	122
9.1	form 对象	122
9.1.1	作为对象和容器的表单	122
9.1.2	访问表单属性	123

9.1.3	form.elements[]属性	123
9.2	作为对象的表单控件	124
9.2.1	与文本相关的对象	124
9.2.2	按钮对象	126
9.2.3	复选框对象	126
9.2.4	单选按钮对象	127
9.2.5	select 对象	128
9.3	向函数传递表单数据和元素	130
9.4	提交和预验证表单	132
第 10 章	用脚本编写框架和多窗口	134
10.1	框架: 父框架和子框架	134
10.2	在家庭成员之间引用	135
10.2.1	父到子的引用	135
10.2.2	子到父的引用	136
10.2.3	子到子的引用	136
10.3	控制多框架	136
10.4	窗口引用的其他内容	138
第 11 章	图像和动态 HTML	141
11.1	image 对象	141
11.1.1	可互换的图像	141
11.1.2	预缓存图像	141
11.1.3	创建图像滚动	143
11.2	javascript: pseudo-URL	146
11.3	常用的动态 HTML 技术	147
11.3.1	更改样式表设置	147
11.3.2	通过 W3C DOM 节点动态处理网页内容	147
11.3.3	通过 innerHTML 属性动态处理网页内容	148
第 3 部分	文档对象模型参考	149
第 12 章	文档对象模型基础	150
12.1	对象模型层次	150
12.1.1	作为路径图的层次	150
12.1.2	浏览器文档对象路径图	151
12.2	文档对象的产生过程	152
12.3	对象属性	152
12.4	对象方法	153
12.5	对象事件处理程序	153
12.5.1	作为方法的事件处理程序	154
12.5.2	作为属性的事件处理程序	154
12.6	对象模型概述	155
12.6.1	基本对象模型	156
12.6.2	基本附加图像对象模型	156
12.6.3	NN4 扩展	156

12.6.4	IE4 扩展	158
12.6.5	IE5 扩展	160
12.6.6	W3C DOM	161
12.7	新 DOM 概念	165
12.7.1	元素引用	165
12.7.2	节点层次	166
12.7.3	节点属性	167
12.7.4	节点方法	169
12.7.5	生成新节点内容	170
12.7.6	替换节点内容	171
12.8	混合对象模型	172
12.8.1	传统的方法	172
12.8.2	中间层	173
12.8.3	改进方法	176
12.8.4	处理事件	177
12.9	标准兼容性模型	177
第 13 章	通用 HTML 元素对象	179
13.1	通用对象	179
13.1.1	语法	181
13.1.2	关于这些对象	181
13.1.3	属性	181
13.1.4	方法	222
13.1.5	事件处理程序	259
第 14 章	window 对象和 frame 对象	288
14.1	window 对象概述	288
14.2	框架	289
14.2.1	创建框架	289
14.2.2	框架对象模型	289
14.2.3	引用框架	290
14.2.4	top 和 parent	291
14.2.5	禁止载入框架	291
14.2.6	允许载入框架	291
14.2.7	从有框架转换到无框架	292
14.2.8	空白框架	292
14.2.9	查看框架源代码	293
14.2.10	框架和 frame 元素对象	293
14.3	window 对象	295
14.3.1	语法	295
14.3.2	属性	320
14.3.3	方法	359
14.3.4	事件处理程序	364
14.4	frame 元素对象	364
14.4.1	语法	365
14.4.2	属性	365

14.5	frameset 元素对象	369
14.5.1	语法	369
14.5.2	属性	370
14.6	iframe 元素对象	373
14.6.1	语法	373
14.6.2	属性	373
14.7	popup 对象	376
14.7.1	语法	377
14.7.2	属性	377
14.7.3	方法	377
第 15 章	location 对象和 history 对象	380
15.1	location 对象	380
15.1.1	语法	381
15.1.2	属性	382
15.1.3	方法	390
15.2	history 对象	392
15.2.1	语法	393
15.2.2	属性	393
15.2.3	方法	394
第 16 章	document 对象和 body 元素对象	398
16.1	document 对象	398
16.1.1	语法	401
16.1.2	属性	401
16.1.3	方法	429
16.1.4	事件处理程序	444
16.2	body 元素对象	445
16.2.1	语法	446
16.2.2	属性	446
16.2.3	方法	450
16.2.4	事件处理程序	451
16.3	TreeWalker 对象	452
16.3.1	语法	452
16.3.2	属性	452
16.3.3	方法	453
第 17 章	link 对象和 anchor 对象	455
17.1	语法	456
17.2	属性	456
第 18 章	image 对象、area 对象和 map 对象	461
18.1	image 对象和 img 元素对象	461
18.1.1	语法	462
18.1.2	属性	462
18.1.3	事件处理程序	473

18.2	area 元素对象	474
18.2.1	语法	475
18.2.2	属性	475
18.3	map 元素对象	477
18.3.1	语法	477
18.3.2	属性	478
第 19 章	form 及其相关对象	480
19.1	form 对象	480
19.1.1	语法	481
19.1.2	引用表单控件	481
19.1.3	将表单和元素传递到函数	481
19.1.4	通过 E-mail 发送表单	484
19.1.5	改变表单属性	485
19.1.6	表单按钮	485
19.1.7	提交后的重定位	485
19.1.8	表单元素数组	486
19.1.9	input 元素对象	486
19.1.10	属性	486
19.1.11	方法	490
19.1.12	事件处理程序	491
19.2	fieldset 和 legend 元素对象	493
19.3	label 元素对象	494
19.3.1	语法	494
19.3.2	属性	495
第 20 章	按钮对象	496
20.1	button 元素对象和 button, submit, reset 输入对象	496
20.1.1	语法	497
20.1.2	属性	497
20.1.3	方法	498
20.1.4	事件处理程序	498
20.2	复选框输入对象	500
20.2.1	语法	500
20.2.2	属性	501
20.2.3	方法	503
20.2.4	事件处理程序	503
20.3	单选按钮输入对象	505
20.3.1	语法	506
20.3.2	属性	506
20.3.3	方法	508
20.3.4	事件处理程序	509
20.4	图像输入对象	510
20.4.1	语法	510
20.4.2	属性	511

第 21 章 与文本相关的表单对象	512
21.1 text 输入对象	512
21.1.1 语法	513
21.1.2 文本域和事件	513
21.1.3 属性	515
21.1.4 方法	519
21.1.5 事件处理程序	521
21.2 password 输入对象	523
21.3 hidden 输入对象	523
21.4 textarea 元素对象	524
21.4.1 语法	525
21.4.2 文本域中的回车符	525
21.4.3 属性	526
21.4.4 方法	526
第 22 章 select, option 和 fileUpload 对象	527
22.1 select 元素对象	527
22.1.1 语法	528
22.1.2 修改 select 选项 (NN3+ 和 IE4+)	529
22.1.3 修改 select 选项 (IE4+)	532
22.1.4 修改 select 选项 (W3C DOM)	533
22.1.5 属性	534
22.1.6 方法	540
22.1.7 事件处理程序	541
22.2 option 元素对象	542
22.2.1 语法	543
22.2.2 属性	543
22.3 optgroup 元素对象	543
22.3.1 语法	544
22.3.2 属性	544
22.4 file 输入元素对象	545
第 23 章 event 对象	547
23.1 事件	547
23.1.1 事件的内容和事件发生的时间	547
23.1.2 静态 Event 对象	548
23.2 事件传播	548
23.2.1 NN4 事件传播	548
23.2.2 IE4+ 事件传播	557
23.2.3 W3C 事件传播	560
23.3 引用 event 对象	564
23.3.1 IE4+ 中的 event 对象引用	565
23.3.2 W3C 中的 event 对象引用	565
23.4 event 对象兼容性	566
23.5 深入了解事件模型	567

23.5.1	检查跨平台的修饰键	567
23.5.2	跨平台的按键捕获	568
23.6	事件类型	569
23.6.1	早期浏览器中的事件类型	569
23.6.2	IE4+ 和 NN6+/W3C 中的事件类型	570
23.7	NN4 中的 event 对象	571
23.7.1	语法	571
23.7.2	属性	572
23.8	IE4+ 中的 event 对象	576
23.8.1	语法	576
23.8.2	属性	577
23.9	NN6+/Moz/Safari 中的 event 对象	589
23.9.1	语法	590
23.9.2	属性	590
23.9.3	方法	602
第 24 章	主体文本对象	604
24.1	blockquote 元素对象和 q 元素对象	604
24.1.1	语法	604
24.1.2	属性	604
24.2	br 元素对象	605
24.2.1	语法	605
24.2.2	属性	605
24.3	font 元素对象	605
24.3.1	语法	605
24.3.2	属性	606
24.4	h1~h6 元素对象	608
24.4.1	语法	608
24.4.2	属性	608
24.5	hr 元素对象	608
24.5.1	语法	609
24.5.2	属性	609
24.6	label 元素对象	611
24.6.1	语法	611
24.6.2	属性	612
24.7	marquee 元素对象	612
24.7.1	语法	613
24.7.2	属性	613
24.7.3	方法	616
24.7.4	事件处理程序	616
24.8	Range 对象	617
24.8.1	语法	618
24.8.2	使用 range 对象	619
24.8.3	属性	619
24.8.4	方法	621
24.9	selection 对象	631

24.9.1	语法	632
24.9.2	属性	632
24.9.3	方法	634
24.10	Text 和 TextNode 对象	636
24.10.1	语法	637
24.10.2	属性	637
24.10.3	方法	637
24.11	TextRange 对象	640
24.11.1	语法	641
24.11.2	使用 TextRange 对象	642
24.11.3	浏览器兼容性	642
24.11.4	属性	643
24.11.5	方法	645
24.12	TextRectangle 对象	658
24.12.1	语法	659
24.12.2	属性	659
第 25 章	表和列表对象	661
25.1	表对象层次结构	661
25.1.1	填充单元格	662
25.1.2	修改单元格内容	663
25.1.3	修改表行	665
25.1.4	修改表列	670
25.1.5	W3C DOM 表对象类	672
25.2	table 元素对象	673
25.2.1	语法	674
25.2.2	属性	674
25.2.3	方法	681
25.3	tbody, tfoot 和 thead 元素对象	683
25.3.1	语法	684
25.3.2	属性	684
25.4	caption 元素对象	685
25.5	col 和 colgroup 元素对象	685
25.5.1	语法	686
25.5.2	属性	686
25.6	tr 元素对象	687
25.6.1	语法	688
25.6.2	属性	688
25.6.3	方法	689
25.7	td 和 th 元素对象	689
25.7.1	语法	690
25.7.2	属性	690
25.8	ol 元素对象	692
25.8.1	语法	692
25.8.2	属性	692
25.9	ul 元素对象	694

25.9.1	语法	694
25.9.2	属性	694
25.10	li 元素对象	694
25.10.1	语法	695
25.10.2	属性	695
25.11	dl, dt 和 dd 元素对象	695
25.12	dir 和 menu 元素对象	696
第 26 章	样式表和样式对象	697
26.1	理解对象名称	697
26.2	导入样式表	698
26.3	style 元素对象	698
26.3.1	语法	699
26.3.2	属性	699
26.4	styleSheet 对象	699
26.4.1	语法	700
26.4.2	属性	700
26.4.3	方法	704
26.5	cssRule 和 rule 对象	705
26.5.1	语法	706
26.5.2	属性	706
26.6	currentStyle, runtimeStyle 和 style 对象	707
26.6.1	语法	708
26.6.2	Style 属性	708
26.6.3	属性值	709
26.6.4	text 和 font 属性	711
26.6.5	inline display 和 layout 属性	716
26.6.6	positioning 属性	720
26.6.7	background 属性	722
26.6.8	border 和 edge 属性	723
26.6.9	list 属性	726
26.6.10	scrollbar 属性	726
26.6.11	table 属性	726
26.6.12	page 和 printing 属性	727
26.6.13	miscellaneous 属性	728
26.7	filter 对象	729
26.7.1	语法	729
26.7.2	读 / 写子属性	729
26.7.3	IE5.5+ 滤镜语法变化	733
第 27 章	navigator 和其他环境对象	738
27.1	clientInformation 对象和 navigator 对象	738
27.1.1	语法	739
27.1.2	属性	739
27.1.3	方法	749
27.2	mimeType 对象	752