

TP325.4
v4.5C

McGraw-Hill

科海电脑技术丛书

JavaScript示例导学

[美] Martin Webb,
Michel Plungjan 著
& Keith Drakard

聊宏斌 胡燕荣 译

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载,
也可到视听部复制

清华大学出版社

<http://www.tup.tsinghua.edu.cn>

(京)新登字158号

著作权合同登记号: 01-2001-3286

内 容 提 要

本书通过 120 多个精彩示例全面讲述了学用 JavaScript 进行编程的方法和技巧,通过对示例代码的解析,使读者在编程过程中体会 JavaScript 易学易用、功能强大的特性。

全书内容包括:JavaScript 基础知识;使用 JavaScript 开发交互式 Web 站点、创建计算器、计算日期和时间、进行密码验证/保护、建立图像映射、创建购物车和联机目录及编写图标匹配、俄罗斯方块、拼色成行等趣味程序;同时讲述了在不同环境下如何定制和扩展脚本程序,及有关 DHTML、插件、安全问题等方面的内容。

本书内容全面,趣味性强,适用于 JavaScript 初级用户,也适用于自学或对网页编程感兴趣的读者。

Instant JavaScript

Copyright © 2001 by The McGraw-Hill Companies.

Simplified Chinese translation edition jointly published by McGraw-Hill Education (Asia) Co., Tsinghua University Press, and Beijing KeHai Training Center Technology Ltd.

本书中文简体字版由清华大学出版社、北京科海培训中心和美国 McGraw-Hill 教育(亚洲)出版公司合作出版。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

版权所有,盗版必究。

本书封面贴有 McGraw-Hill 公司激光防伪标签,无标签者不得销售。

书 名: JavaScript 示例导学

作 者: Martin Webb, Michel Plungjan & Keith Drakard

译 者: 聊宏斌 胡燕荣

出版者: 清华大学出版社(北京清华大学校内,邮编 100084)

印刷者: 北京市耀华印刷有限公司

发行者: 新华书店总店北京科技发行所

开 本: 787×1092 1/16 印张: 33.875 字数: 824 千字

版 次: 2002 年 4 月第 1 版 2002 年 4 月第 1 次印刷

印 数: 0001~4000

盘 号: ISBN 7-900641-72-6

定 价: 54.00 元(光盘)

内容介绍

本书内容分为两部分。第1部分主要介绍各种JavaScript组件，包括从基本数学表达式到利用Cookie保存网页的稳定状态等内容。该部分中的每一章都会详细介绍关于使用组件的各种情况，并通过列举一些实用性强，并且既常用又可重用的应用程序实例来说明如何使用组件。第1部分包括如下章节：

第2章：数学与数字

本章描述了用JavaScript编写的两个计算器：标准计算器与科学计算器，并且介绍了如何利用随机数生成器来显示随机名言警句。本章介绍的内容还包括一些数字工具函数，它们在本书其他章节中多次重用。

第3章：字符串和正则表达式

本章介绍如何查找与替换字符串，并提供一个带有模板的实用性范例，可利用该模板简便地生成版权信息。本章还介绍了如何创建滚动文本链接、如何生成联系列表、以及如何操纵应用高德阴影处理技术的字符串。本章还描述了如何分析文本字符串，最后介绍如何使用正则表达式来验证表单的数据输入。

第4章：数组和对象

本章介绍如何将数据保存为对象、如何将对象保存为数组，以及如何访问自动漫游的多个站点中的一个。本章还介绍如何查找数据库中的数据，以及如何查找多项选择测试。最后，解释如何对数组中的对象进行排序。

第5章：定位和链接

本章描述如何与当前定位进行交互操作，并进一步提供对新位置的自动重定向。通过对不同页面间进行自动查找前一页和后一页的按钮操作的解释，以及同一页上不同部分间自动链接的解释，描述如何利用JavaScript管理相关的网站链接。本章还介绍了如何获取浏览器所指定的语言，以及如何载入相应的网页。另外，本章也提供了如何保护密码和验证输入密码的相关信息。最后，介绍了如何描写JavaScript链接爬行器来查找所有相关页面中的链接。

第6章：窗口和框架

本章描述如何将选择表格列表中的条目定位到框架集合中的不同窗口，解释了如何读出目录列表中的条目，并描述了如何生成可以自弹出窗口的工具，对如何使用查找字典的弹出窗口也有所描述；同时，还讲述了如何利用框架来获得页面视图间的数据并生成家族树的应用。在本章中，还解释了如何使用弹出式窗口来调试自己的应用程序，其后是弹出

式日期选择器应用，最后利用代码来防止其他站点对你的页面的框架化动作，同时提供一种方法能够确保你的页面在框架集合中能够正确形成框架。

第 7 章：日期和时间

本章介绍了如何创建和维护当前日期与时间。介绍了如何从任何日期中计算出星期几，如何控制输入数据，以及如何按照时间显示不同的页面。还描述了如何显示年历，如何向前和向后计算不同的日期，如何按照年月日来计算年龄，如何计算星座与农历。本章包括与日期有关的应用程序，例如事件、宗教节日、生日等。最后介绍如何计算出世界各地的当前时间。

第 8 章：表单

本章第一次描述了HTML表格如何与JavaScript交互，其中包括了相关应用程序，它介绍了如何发送电子邮件表单，如何与下拉式菜单一起实现交互操作，如何将数据从一个表单传递到另一个表单，还包括如何提供带有动态提交表单的感谢页面，以及如何通过Cookies和电子邮件将数据从一个表单传递到另一个表单的应用程序。本章还描述了如何在HTML 4中禁用表单域，包括可以控制表单数据条目的技术，也包括如何过滤输入数据，以及如何生成新的表单组件来调整日期。最后，本章解释如何提供订单，以计算各种必须的税率、小计及总计。

第 9 章：图像

本章包括广泛使用的图像高亮显示、快速图像滚动、图像淡入淡出等应用。本章举例说明如何使用图像映射的应用，而该应用可以显示幻灯片，也能生成条形图。还介绍了如何计算并显示下载图像的数目，如何弹出窗口保存图像，其中包括图像计数器，随机标题广告，以及图像工具栏等。

第 10 章：Cookies

本章描述常用的Cookie函数，如保存、检索和删除Cookie，并显示了如何将Cookie和HTML组合在一起更新条目标志。对于有一定限制的浏览器，本章对避免Cookie连续写的智能Cookie也有所涉及。本章的Cookie跟踪器应用示范了如何将Cookie用于跟踪访问者在网站间的移动。而个人日历应用则显示如何利用Cookie保存个人化的信息。

本书第2部分内容主要介绍在前一部分内容的基础上如何创建更为复杂的应用程序，同时建立了一个多种浏览器可用的代码库，它可以在Netscape Navigator（版本4/6）和Internet Explorer（版本4/5）中支持动态HTML技术。这部分内容包括以下章节：

第 11 章：动态 HTML

本章描述的文档对象模型（DOM）包括了Internet Explorer 5和Netscape Navigator 6 中可用的模型。为了操作DOM，创建了多种浏览器兼容的数据库，还示范了如何使用此数据库来压缩内容，如何动态改变层叠样式表，这些样式表可以用于显示页面，显示在文档上浮动的工具提示，处理多对话框表单以及具有下拉内容的工具栏菜单。该数据库也可用

于从文档的标题标签中动态生成目录，从而生成一个可供打开和关闭的层次菜单，这样就可以通过选中它们察看文档的相应内容。最后该应用示范了可以与用户交互的动态广告。

第 12 章：应用

本章介绍一些初具规模的应用程序，包括购物筐应用程序的两个版本，版本一用简单的超文本链接和表单按钮实现与用户的交互，而版本二则使用拖放技术实现该功能。myPage.com应用显示了用户如何将其网页的浏览方式进行一种个性化处理。计算机建造器/选择器示范了如何从简单的产品与组件数据库中生成多种产品的不同复杂视图。书籍分类应用程序则显示了JavaScript如何与XML文档、DOM、样式表等在一个应用中进行交互。

第 13 章：工具

本章包括JavaScript的多种用法。拾色器允许作者实验各种背景与前景的混合颜色。通过屏上键盘，用户可以与页面进行交互，甚至在键盘不存在的情况下也可以进行交互。自动滚屏应用程序说明了在没有用户干涉的情况下，页面是如何滚动的。主页脚本用于选择当前页为用户默认主页。最后的应用程序显示了如何检测对插件和ActiveX组件的支持，它们都对基本浏览器的功能进行了扩展。

第 14 章：趣味内容

最后一章介绍了JavaScript的很多有趣味的功能，包括三个JavaScript游戏：从简单的图标匹配游戏到更有趣的俄罗斯方块和堆控制的JavaScript版本。有趣的鼠标踪迹应用显示了如何使用DHTML技术跟踪屏幕上的鼠标位置。本书最后的应用是个性测试，可用于评定用户的个性或者缺点。

关于配套光盘

书中所有程序代码都附在光盘中。

如果光盘中的代码有问题或本书有任何遗漏，请与马丁（martin@irt.org.）联系，若您对以后版本有好的建议，也请与我们联系。任何在印刷后的更新或者勘误可在下面网站中查找到详细信息：<http://www.irt.org/instant/>。

目 录

第1章 JavaScript概述	1
1.1 JavaScript、JScript、DHTML与DOM	1
1.1.1 JavaScript标准	1
1.1.2 在哪里放置JavaScript代码	2
1.1.3 只在客户端JavaScript中出现的问题	5
1.1.4 JavaScript版本	5
1.2 JavaScript基础	7
1.2.1 表达式与变量	7
1.2.2 语句	7
1.2.3 条件语句	9
1.2.4 循环	10
1.2.5 函数	11
1.2.6 对象	12
1.2.7 对象属性	14
1.2.8 对象方法	14
1.2.9 事件与事件处理函数	15
1.2.10 双引号与单引号	16
1.2.11 noscript标记	16
1.3 小结	17
第2章 数学与数字	18
2.1 标准计算器	18
2.2 科学计算器	30
2.3 随机每日名言警句	45
2.3.1 随机名言警句源代码	47
2.3.2 JavaScript quote1.js源代码	48
2.4 数字工具	49
2.4.1 数字舍入到指定小数位	49
2.4.2 显示货币金额	49
2.4.3 用逗号格式化大数	50
2.4.4 用逗号格式化货币金额	50
2.4.5 将99格式化为Ninety-Nine	51
2.4.6 将999格式化为Nine Hundred and Ninety-Nine	51
2.4.7 将大额货币格式化为字符串	51
2.5 小结	53

第3章 字符串和正则表达式	54
3.1 查找和替换	54
3.2 Copyright模板文件	57
3.2.1 Copyright模板文件的源代码	57
3.2.2 字符串方法	59
3.3 滚动文本链接	60
3.4 联系列表	68
3.5 高德阴影	74
3.5.1 字符串高德阴影效果源代码	76
3.5.2 表格高德阴影效果源代码	79
3.6 分析字符串	81
3.7 使用正则表达式验证字符串	82
3.7.1 字符串验证源代码	84
3.7.2 正则表达式	90
3.8 小结	92
第4章 数组和对象	93
4.1 遍历访问Web站点	93
4.1.1 遍历访问Web站点的组成	93
4.1.2 遍历访问Web站点的框架集结构	94
4.1.3 遍历访问Web站点源代码	96
4.2 搜索数据库	101
4.2.1 基于框架的搜索	102
4.2.2 基于框架搜索源代码	103
4.2.3 基于非框架的搜索	110
4.2.4 基于非框架搜索源代码	112
4.2.5 搜索数据库代码的扩展	120
4.3 多项选择测试	121
4.3.1 多项选择测试源代码——questions.js	121
4.3.2 多项选择测试源代码——choice.htm	122
4.4 对象数组排序	126
4.4.1 对象数组排序源代码	128
4.4.2 数组方法	133
4.5 小结	134
第5章 定位和链接	135
5.1 自动重定向	135
5.2 自动的Previous和Next按钮	139
5.2.1 自动的Previous和Next按钮源代码	140
5.2.2 自动的Previous和Next按钮buttons.js源代码	141

5.3 自动的Previous和Next按钮directories.js源代码	142
5.4 自动链接	144
5.4.1 自动链接源代码——文本版	146
5.4.2 自动链接源代码——图像版	148
5.5 JavaScript Web Ring	149
5.5.1 JavaScript Web Ring 源代码——剪切和粘贴HTML	150
5.5.2 JavaScript Web Ring 源代码——webring.js	151
5.6 你的语言是什么	153
5.7 密码保护和验证	159
5.8 密码保护和验证源代码	162
5.9 网站链接爬行器	165
5.9.1 网站链接爬行器源代码——index.htm	168
5.9.2 网站链接爬行器源代码——input.htm	169
5.9.3 网站链接爬行器源代码——frame.htm	177
5.9.4 网站链接爬行器源代码——search.htm	178
5.10 小结	181
第6章 窗口和框架	182
6.1 Jump Box工具	183
6.1.1 Jump Box工具源代码——index.htm	183
6.1.2 Jump Box工具源代码——control.htm	184
6.2 读取目录	187
6.2.1 读取目录的源代码——index2.htm	189
6.2.2 读取目录的源代码——main.htm	189
6.3 弹出式窗口产生器	192
6.3.1 弹出式窗口产生器源代码	194
6.3.2 窗口属性	196
6.4 弹出式字典	197
6.4.1 弹出式字典源代码——index.htm	197
6.4.2 弹出式字典源代码——define.htm	198
6.4.3 弹出式字典源代码——外部JavaScript源文件	200
6.5 家族树	201
6.6 家族树源代码——royal.js	210
6.7 家族树源代码——topframe.htm	212
6.8 家族树源代码——botframe.htm	212
6.9 一个JavaScript调试窗口	213
6.10 弹出式日期选择器	219
6.10.1 弹出式日期选择器源代码——index.htm	220
6.10.2 弹出式日期选择器源代码——cal.htm	222

6.10.3 弹出式日期选择器源代码——calendar.js.....	223
6.11 取消框架化和重新框架化.....	223
6.11.1 取消框架化和重新框架化源代码——frameset.htm.....	224
6.11.2 取消框架化和重新框架化源代码——main.htm.....	227
6.11.3 取消框架化和重新框架化源代码——title.htm.....	229
6.12 小结.....	229
第7章 日期和时间.....	230
7.1 当前日期和时间.....	231
7.1.1 当前日期和时间源代码——current.js.....	233
7.1.2 当前日期和时间源代码——index.htm.....	236
7.2 Monday's Child Is Fair of Face.....	236
7.2.1 Monday's Child Is Fair of Face源代码——Monday.js.....	237
7.2.2 Monday's Child Is Fair of Face源代码——index.htm.....	238
7.3 有效日期.....	238
7.3.1 有效日期源代码——validate.js.....	239
7.3.2 有效日期源代码——index.htm.....	241
7.4 显示时间.....	243
7.4.1 显示时间源代码——form.htm.....	243
7.4.2 显示时间源代码——nightandday.htm.....	245
7.4.3 显示时间源代码——timeofday.js.....	246
7.4.4 显示时间源代码——image.htm.....	248
7.5 年历.....	249
7.5.1 年历源代码——calendar.js.....	250
7.5.2 年历源代码——calendar.css.....	256
7.5.3 年历源代码——index.htm.....	256
7.6 日期倒计时.....	258
7.6.1 日期倒计时源代码——countdown.js.....	259
7.6.2 日期倒计时源代码——index.htm.....	260
7.7 32岁8个月24天.....	261
7.7.1 32岁8个月24天源代码——howold.js.....	262
7.7.2 32岁8个月24天源代码——index.htm.....	263
7.8 你的星座是什么.....	264
7.8.1 你的星座是什么源代码——zodiac.js.....	264
7.8.2 你的星座是什么源代码——index.htm.....	266
7.9 中国春节.....	267
7.9.1 中国春节源代码——chinese.js.....	268
7.9.2 中国春节源代码——index.htm.....	271
7.10 生于7月4日.....	273

7.10.1 生于7月4日源代码——july4th.js.....	273
7.10.2 生于7月4日源代码——index.htm.....	275
7.10.3 生于7月4日源代码——birthjan.js.....	277
7.11 11月的第3个星期六.....	277
7.11.1 11月的第3个星期六源代码——Saturday.js.....	278
7.11.2 11月的第3个星期六源代码——index.htm.....	278
7.12 计算复活节.....	280
7.12.1 计算复活节源代码——easter.js.....	281
7.12.2 计算复活节源代码——index.htm.....	284
7.13 24小时时钟.....	285
7.14 小结.....	288
第8章 表单.....	289
8.1 JavaScript和表单.....	289
8.2 从表单中发送邮件.....	293
8.2.1 从表单中发送邮件源代码——simple.htm.....	294
8.2.2 从表单中发送邮件源代码——multiple.htm.....	295
8.3 动态下拉菜单.....	297
8.3.1 动态下拉菜单源代码——single.htm.....	298
8.3.2 动态下拉菜单源代码——multiple.htm.....	300
8.3.3 动态下拉菜单源代码——transfer.htm.....	301
8.3.4 动态下拉菜单源代码——colors.htm.....	302
8.4 将数据从一个表单传递到另一个表单.....	304
8.4.1 将数据从一个表单传递到另一个表单源代码——index.htm.....	305
8.4.2 将数据从一个表单传递到另一个表单源代码——output.htm.....	306
8.5 动态感谢页面.....	307
8.5.1 动态感谢页面源代码——index.htm.....	308
8.5.2 动态感谢页面源代码——userform.cgi.....	308
8.5.3 动态感谢页面源代码——thanks.htm.....	310
8.6 表单->Cookie->表单.....	311
8.7 表单->电子邮件->表单.....	316
8.7.1 表单->电子邮件->表单源代码——index.htm.....	316
8.7.2 表单->电子邮件->表单源代码——mailform.js.....	317
8.8 禁用表单域.....	321
8.9 使用表单域控制数据输入.....	323
8.10 过滤输入数据.....	325
8.11 日期微调控制项.....	327
8.11.1 日期微调控制项源代码——index.htm.....	327
8.11.2 日期微调控制项源代码——index2.htm.....	329

8.12 订购单表单	329
8.13 小结	332
第9章 图像	333
9.1 高亮图像	333
9.2 快速图像滚动	336
9.3 图像淡入淡出	339
9.4 图像映射	342
9.5 幻灯片显示	347
9.6 条形图	351
9.6.1 条形图源代码——barchart.js	352
9.6.2 条形图源代码——index.htm	356
9.7 图像下载百分比	358
9.8 图像弹出	361
9.8.1 图像弹出源代码——index.htm	362
9.8.2 图像弹出源代码——popup.htm	364
9.8.3 图像弹出源代码——resize.htm	365
9.9 图像计数器	366
9.10 随机标题广告	369
9.11 图像工具栏	372
9.11.1 图像工具栏#1源代码——section-a.htm	373
9.11.2 图像工具栏#1源代码——section-b.htm	373
9.11.3 图像工具栏#2源代码——section-a.htm	374
9.11.4 图像工具栏#2源代码——toolbar.js	374
9.12 小结	376
第10章 Cookies	377
10.1 JavaScript Cookie通用例程	378
10.1.1 JavaScript Cookie通用例程源代码——index.htm	378
10.1.2 JavaScript Cookie通用例程源代码——cookie.js	380
10.2 Cookie与自动更新	381
10.3 智能Cookie	384
10.3.1 智能Cookie源代码——index.htm	385
10.3.2 智能Cookie源代码——intelligent.js	386
10.4 Cookie跟踪器	388
10.4.1 Cookie跟踪器源代码——index1.htm	389
10.4.2 Cookie跟踪器源代码——index2.htm	390
10.4.3 Cookie跟踪器源代码——tracker.js	390
10.4.4 Cookie跟踪器源代码——tracking.js	391
10.5 个人日历	392

10.5.1 个人日历源代码——index.htm.....	393
10.5.2 个人日历源代码——calendar.js	395
10.5.3 个人日历源代码——calendar.css	395
10.5.4 个人日历源代码——cookie.htm.....	396
10.6 小结	398
第11章 动态HTML.....	399
11.1 文档对象模型接口	400
11.2 可折叠内容	403
11.3 动态样式表	406
11.3.1 动态样式表源代码——index.htm.....	407
11.3.2 动态样式表源代码——stylesheet.js	408
11.3.3 动态样式表源代码——next.htm.....	409
11.4 浮动工具提示	409
11.5 多对话框表单	412
11.6 工具栏菜单	416
11.7 自动生成目录	420
11.8 标签幻灯片	424
11.9 菜单系统	428
11.9.1 菜单系统源代码——menu.js	428
11.9.2 菜单系统源代码——index.htm.....	430
11.10 动态广告	431
11.11 小结	434
第12章 应用	435
12.1 购物筐	435
12.1.1 购物筐源代码——shopping.js	436
12.1.2 购物筐源代码——cookie.js	440
12.1.3 购物筐源代码——index.htm.....	441
12.1.4 购物筐源代码——basket.htm	441
12.2 拖拉式购物车	442
12.3 myPage.com	449
12.4 在线计算机建造器/选择器	455
12.4.1 在线计算机建造器/选择器源代码——builder.js.....	456
12.4.2 在线计算机建造器/选择器源代码——database.js	461
12.4.3 在线计算机建造器/选择器源代码——index.htm.....	463
12.4.4 在线计算机建造器/选择器源代码——windows.js.....	463
12.4.5 在线计算机建造器/选择器源代码——choice.htm	465
12.4.6 在线计算机建造器/选择器源代码——confirm.htm	467
12.4.7 在线计算机建造器/选择器源代码——money.js	470

12.5 书籍分类——XML DOM和样式表	470
12.5.1 书籍分类源代码——catalog.xml	471
12.5.2 书籍分类源代码——catalog.js	473
12.5.3 书籍分类源代码——catalog1.css	476
12.5.4 书籍分类源代码——catalog2.css	476
12.6 小结	477
第13章 工具	478
13.1 拾色器	478
13.1.1 拾色器源代码——index.html	478
13.1.2 拾色器源代码——picker.htm	480
13.2 屏上键盘	482
13.3 自动滚屏	486
13.4 主页脚本	487
13.4.1 主页脚本源代码——index.htm	488
13.4.2 主页脚本源代码——homepage.js	489
13.5 探测插件和ActiveX组件	490
13.5.1 检查plugins数组	491
13.5.2 探测ActiveX组件	494
13.5.3 探测插件和ActiveX组件源代码——index.htm	495
13.6 小结	497
第14章 趣味内容	498
14.1 图标匹配	498
14.2 图标匹配源代码——index.htm	498
14.3 俄罗斯方块	504
14.4 拼色成行	515
14.5 鼠标踪迹	521
14.6 个性测试	524
14.6.1 个性测试源代码——index.htm	524
14.6.2 个性测试源代码——personality.js	525
14.6.3 个性测试源代码——questions.js	526
14.7 小结	528

第1章 JavaScript概述

1.1 JavaScript、JScript、DHTML与DOM

早在1995年，Brendan Eich就创建了JavaScript，并将它用于Netscape Navigator 2中。它客户端浏览器可以动态地生成HTML文本，并允许其执行JavaScript代码，产生可供用户查看并与之交互的页面。嵌在HTML页面中的JavaScript代码可以响应用户事件，如鼠标单击、表单输入和页面导航。JavaScript这个术语专指Netscape Navigator浏览器中的Netscape实现，而JScript这个术语则专指Microsoft在Internet Explorer中实现的JavaScript。本书既包括Netscape的JavaScript，也包括Microsoft的JScript，它们分别用于Netscape 和IE。本书其他部分会用JavaScript这个术语指代上述两者。

动态HTML（DHTML），尽管不是一个明确定义的术语，但大多数人都接受用这个术语来描述HTML、JavaScript、文档对象模型（DOM）、层叠样式表（CSS）的组合。本书11章“动态HTML”详细介绍了DHTML，而且第12章“应用”则通过DHTML 应用程序进一步支持DHTML。WWW联盟（W3C）声称“文档对象模型是平台无关、语言无关的接口，允许程序和脚本动态地访问和更新文档的内容、结构与风格。”

JavaScript不是Java，同样，Java也不等同于JavaScript。JavaScript是独立于Java开发的。起初所开发的是一种叫做LiveScript的产品，但在Netscape宣布Netscape Navigator 2.0支持Java之后，它就改名换姓了。JavaScript是一种脚本语言，通常是由浏览器中的JavaScript解释器在运行时解释执行的。代码可以嵌入在HTML页面中，或是作为一个外部的JavaScript文件独立存在。而Java与此截然不同，它首先被编译成字节代码，然后在Java虚拟机（JVM）上运行，可以独立运行，当然也可以在浏览器中运行。尽管JavaScript被描述为一种面向对象的语言，但它并不是一种纯粹的、百分之百的面向对象语言。而Java则是一种纯粹的面向对象语言。

尽管两者的语法不完全一样，但它们彼此却颇为相似。在Netscape Navigator中，JavaScript可以激活Java代码——这一功能叫做LiveConnect。不管是Java还是LiveConnect，都不是本书的讨论范围。

1.1.1 JavaScript标准

欧洲标准组织ECMA公布了ECMAScript语言规范。其首次发行版本是基于Netscape的JavaScript 1.1实现完成的。目前最新的发行版本是版本3，是基于1999年12月公布的Netscape 的JavaScript 1.5 实现完成的。它既有Microsoft的Word文档格式（<http://www.mozilla.org/js/language/E262-3.doc>），也有Acrobat的PDF文件格式（<http://www.mozilla.org/js/language/E262-3.pdf>）。ECMA-262 已经被ISO/IEC JTC 1接受，作为ISO/IEC 16262。ECMAScript是一种语言规范，因此它本身并没有定义对象模型

（除了一个“本地对象”集合）。

脚本语言中使用的 Web 浏览器对象模型，或所谓 DOM，已由 W3C（<http://www.w3.org>）进行了标准化。它存在三个 DOM 级别：级别 0 的 DOM 是在 Netscape Navigator 3.x 和 Internet Explorer 3.x 中提供的对象模型；级别 1 的 DOM 是目前 W3C 所推荐的（<http://www.w3.org/TR/REC-DOM-Level-1>）；级别 2 的 DOM 也是目前 W3C 所推荐的（<http://www.irt.org/TR/>）。目前，W3C 正在起草第 3 级别 DOM 的草案。

除了 ECMAScript 和 W3C 文档外，可能最有用的文档是浏览器供应商自己的 JavaScript 文档。JScript 的文档可在 Microsoft 的站点上获得 <http://msdn.microsoft.com/scripting/default.htm?scripting/jscript/default.htm>；JavaScript 的文档可在 Netscape 站点上获得：<http://developer.netscape.com/tech/javascript/index.html>

1.1.2 在哪里放置 JavaScript 代码

JavaScript 在页面中到底放在哪里呢？通常，在人们第一次接触到 JavaScript 时，首先都会问到这个问题。简单的回答是：JavaScript 是放在一个起始标记和一个结尾标记之间。本书的第一个 JavaScript 例子代码，如下所示：

见光盘：chapter1/examples/sample.htm

```
<html>

<head>
<title>Sample Code</title>
</head>

<body>

<script language="JavaScript"><!--
document.write('Hello World');
//--></script>

</body>

</html>
```

字符 `<!--` 和 `-->` 是 HTML 的起始标记（`<!`）结束标记（`-->`）与 JavaScript 的注释符（`//`）的组合，当浏览器不支持 JavaScript 时，这样可以有效地隐藏脚本标记的内容。如果在不支持 JavaScript 的浏览器中遗漏了这些标记，脚本标记的内容将被以普通正文方式显示在页面上。

放置 JavaScript 代码的位置取决于要让 JavaScript 代码作什么。如果只是在加载或处理页面时需要用 JavaScript 代码来简单影响页面，比如上面的例子，那么仅需要将 JavaScript 代码放置在页面体的脚本元素间。但是，如果要定义一个可重用的 JavaScript 代码块，并在页面的生存期内多次使用，那么则要将 JavaScript 代码放在函数块中是在文档头中的脚本标记中声明。如果需要 JavaScript 对用户触发的事件进行反应，那么还有一种可能就是要将 JavaScript 代码放在合适的 HTML 标记的 HTML 属性中，在函数块中的 JavaScript 代码也

支持这种放置方法。另外，本书广泛使用的方法就是将可重用的JavaScript代码放在外部JavaScript源代码库中。上述所有这些可能性都会有进一步的解释和说明。

我们建议，任何没有写到文档中的JavaScript代码都应该放在文档头部。下面的例子定义了一个代码块，其实只有一行代码，即`functionName()`。调用该代码的方法就是直接引用它的名字，其后跟一对空括号就可以了。名为`test`的JavaScript变量被定义为字符串，并初始化为字符串`Hello World`。这些行为不可能影响到页面的显示，所以将它们放置在文档头部是很合适的。

在这个例子中，另一处使用JavaScript代码的作用是在文档体元素标记中生成HTML标记。这种使用JavaScript的方法更好地运用了JavaScript各个方面的功能（例如，基于运行时配置生成HTML标记——比如，正在使用的浏览器，当天的时间，以及通过URL传递的信息等就是这类运行时配置信息）。

注意：由于在页面被显示出来的时候，JavaScript代码是将内容写在页面上的，所以必须在页面完成加载之前执行代码。否则，如果页面已经下载，利用JavaScript写文档，那么当前页面、HTML标记、以及正在执行的JavaScript代码都会被最新产生的输出所替代。也许这种状况正是所需要的现象，但通常情况下，这又是那些不熟悉JavaScript的人最先遇到的问题。

JavaScript的另一种用法是将标记的实际内容写入文档。按钮表单域的`onClick`属性探测到用户单击表单按钮的事件，并通过调用定义在文档头中的`nameFunction()`函数截获该事件：

见光盘：`chapter1/examples/functionName.htm`

```
<html>

<head>

<script language="JavaScript"><!--
function functionName() {
    alert(text);
}

var text = 'Hello World';
//--></script>

</head>

<body>

<script language="JavaScript"><!--
document.write(
    '<form>' +
    '<input type="button" value="Click Here" ' +
    'onClick="functionName()"></form>'
);
```

```
//--></script>

</body>

</html>
```

在文档头中定义JavaScript函数可以确保在需要JavaScript函数以前, 已经将它们成功载入浏览器了。并且如果总能在文档头中找到JavaScript代码, 也更容易进行维护。

JavaScript源文件可以用于Netscape Navigator 3和Internet Explorer 3.01, 而且对于含有JavaScript代码的外部JavaScript文件, 可以被嵌入到一个或多个HTML文件中。这样的好处是, 文件一旦被下载到浏览器的高速缓存中, 对于其他HTML文件而言, 也是可以重用的。

因此, JavaScript源文件成为一种有效的可重用的JavaScript代码库。但是, 在使用HTML文件中的源文件代码前, 应该进行一下测试以确保源文件已经正确下载, 并且它的内容也是可用的。这里, 当页面已被完全加载以后, 会再次使用事件处理程序来调用hello()函数, 但只有library.js的JavaScript源文件已经被载入, 并具有一个名字为hello的属性后, 才可能实现。换句话说, 名为hello()的函数已被声明和定义为当前窗口的属性。

见光盘: chapter1/examples/library.htm

```
<html>

<head>
<title>JavaScript Source File</title>
<script src="library.js"></script>
</head>

<body onLoad="if (window.hello) hello()">
</body>

</html>
```

然后通过下面的JavaScript代码, 就可以简单地将函数定义到library.js的JavaScript 源文件中了, 这里省略了通常的script的起始标识和结尾标记:

见光盘: chapter1/examples/library.js

```
function hello() {
    alert('Hello World');
}
```

注意: 在Microsoft的Internet Explorer 3.01中使用JavaScript源文件时, 必须加以小心。例如使用document.write()将内容写到外部源文件时, 是不会写入文档的, 有时甚至会引起浏览器崩溃。

注意: 如果必须开发代码, 或希望使用本书中介绍的使用JavaScript源文件的应用程序, 对于不支持JavaScript源文件的浏览器而言, 只需要将JavaScript源文件的内容剪切粘贴到文件头中的起始和结尾script标记之间就可以了。