

迅速掌握 Web 设计的宝典秘籍

# JavaScript

## 实例、技巧与应用

蒋民 编著

### what's NEW???

- ◆ JavaScript 简明语法实例
- ◆ 表单技术与 CGI 关系解密、Cookies 使用
- ◆ CSS 与 DHTML 技巧
- ◆ 下拉菜单设计、梳状目录设计、浮动层设计
- ◆ JavaScript 与 Active 技术、数据库访问技术解密
- ◆ 多媒体网页技术
- ◆ 图形特效、鼠标特效、文字特效、各种页面特效技巧介绍



### 光盘内容

- ◆ 书中实例、各类特效实例、各类网页素材



清华大学出版社

# JavaScript 实例、技巧与应用

蒋 民 编著

清 华 大 学 出 版 社  
北 京

## 内 容 简 介

本书全面地介绍了有关 JavaScript 的 Web 设计知识。全书共 9 章，第 1~3 章介绍 JavaScript 的基本编程方法；第 4~6 章介绍 JavaScript 的编程技巧；第 7~8 章介绍一些较为深入的知识，包括面向网站建设的数据库调用、ActiveX 调用以及 ASP 技术等；第 9 章介绍几个 JavaScript 的编程实例。

本书内容充实，实例丰富，适用于初、中级的 Web 设计人员和相关的培训班学员。

版权所有，翻印必究。

本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

书 名：JavaScript 实例、技巧与应用

作 者：蒋 民 编著

出 版 者：清华大学出版社(北京清华大学学研大厦, 邮编 100084)

[http:// www. tup. tsinghua. edu. cn](http://www.tup.tsinghua.edu.cn)

责任编辑：欧振旭

印 刷 者：北京市清华园胶印厂

发 行 者：新华书店总店北京发行所

开 本：787×1092 1/16 印张：18.25 字数：420 千字

版 次：2003 年 4 月第 1 版 2003 年 4 月第 1 次印刷

书 号：ISBN 7-89494-045-3

印 数：0001~4000

定 价：28.00 元(附光盘)

# 前 言

HTML (Hyper Text Markup Language) 的主要功能是指定文章和图怎样在 Internet 浏览器上表示, 但是, 它难以做到在网页浏览器中表示后实现表示位置的移动、数据输入、Event 对应的变化处理等灵活的动作。作为弥补 HTML 这一缺陷的替代, Java 和 JavaScript 赢得了网页设计者的广泛兴趣。Java 是 Sun Microsystems 公司开发的与网络对应的 Object 面向对象编程语言。JavaScript 是 Netscape 公司开发的在 HTML 内基于对象的 Script 编程语言。虽然两者都有“Java”一词, 但是除了都具有 C 语言语系的对象化特征外, 还可以将两者看作完全不同的语言。

在网页设计中, 常常会用到 Fireworks、Dreamweaver、FrontPage 等网页设计工具, 这些软件虽然都能生成 HTML 文件, 而且也可以在 HTML 中加入固定的 JavaScript 程序段实现一些功能, 但要完全了解网页设计的内幕和诀窍, 开发自己的 JavaScript 程序段, 就必须先学习编写 JavaScript 脚本。可以说, JavaScript 的学习是网页设计的核心之一, 在这个基础上再去学习 DHTML、CSS (Cascading Sheet Style)、Java Applet 是比较好的。

JavaScript 在语言风格上有以下特征:

对于许多熟悉 HTML 的初学者而言, 比较难以适应 Java 的编程风格。但使用比较简便的 JavaScript, 可以帮助读者学习理解 Java、DHTML、CSS 的使用。JavaScript 的命令解释器已在 Microsoft Internet Explorer 和 Netscape Navigator 等浏览器内组装, 不需要另外购买安装组件。由于 JavaScript 是依据命令解释器运行, 网页出现错误也可以实时 (Realtime) 表示。因而 JavaScript 对于初学者是非常方便的。

本书将对 JavaScript 分成 9 章来论述, 各章讲解的内容都有具体的例子演示, 可以有效提高学习的效果。读者也可以直接借鉴这些示例编写自己的网页。其中, 第 1~3 章介绍 JavaScript 的基本编程方法; 第 4~6 章介绍 JavaScript 的编程技巧; 第 7~8 章介绍一些深入的知识, 包括面向网站建设的数据库调用、ActiveX 调用以及 ASP 技术; 第 9 章介绍几个 JavaScript 编程的实例。

本书除了提供多个 JavaScript 简明实例外, 还结合实例讲解了 CGI 与表单反馈的关系、当前流行的 CSS 和 DHTML 技巧、极有挑战性的下拉菜单设计、梳状目录设计以及浮动层网页的设计等; 还介绍了具有一定深度的 JavaScript 与 ActiveX 技术, 包括基于 ActiveX 的用户端 Flash 和服务端 ASP 技术; 最令网络程序员头痛的网络数据库调用也在本书中结合 ASP 中 JavaScript 脚本语言做了深入浅出的介绍; 此外还介绍了应用 JavaScript 实现的图形特效、鼠标特效、文字特效等各种页面特效技巧以及多媒体网页技术的实现。本书内容充实, 介绍的许多技术属于网页设计中的前沿技术, 其中许多程序代码经修改后可以直接在自己的网页设计中进行利用。

作者在写作本书的过程中, 阅读了许多国内外的 JavaScript 书籍, 虽然现在已有许多介绍 JavaScript 的书籍, 但大多只是把 JavaScript 作为一种用户端网页脚本语言, 局限于介绍其编程应用方法。像本书这样收集大量实例, 结合目前最为流行的编程技巧和服务器端

技术来作深入浅出介绍的书籍并不多。本书对于网络编程的初学者而言，可以得到大量的实例演示；对于进一步钻研的程序员而言，也可以借鉴本书中目前比较流行的前沿网页技术与技巧实例，如基于 ActiveX 技术的 Flash 动画调用、与 CSS 结合的下拉菜单以及多媒体网页制作等。

本书的写作受到教育部的 IC 网上合作设计项目的支持，在撰稿期间还有幸聆听了北京大学计算机系李晓明教授和吉利久教授的许多讲解和指教，受益匪浅，作者在此深表感谢！由于作者水平有限，错误难免，恳请广大读者提出宝贵意见，共勉共进。

作者于未名湖畔

2003 年 2 月

# 目 录

第 1 章 JavaScript 概要.....	1
1.1 JavaScript 语言概况.....	2
1.2 JavaScript 与 Java 的比较.....	3
1.3 HTML 中的 JavaScript 脚本.....	5
1.4 外部 JavaScript 脚本的做法.....	7
1.5 JavaScript 的相关信息.....	8
第 2 章 JavaScript 主要语言格式.....	11
2.1 JavaScript 的基本数据结构.....	12
2.1.1 常量.....	12
2.1.2 变量.....	13
2.1.3 表达式.....	14
2.1.4 运算符.....	14
2.1.5 实例解说.....	14
2.2 JavaScript 程序流程控制的基本语法.....	17
2.2.1 while 语句和 for 语句.....	17
2.2.2 if 语句.....	19
2.2.3 多重循环.....	22
2.3 JavaScript 程序中的函数调用.....	23
2.3.1 函数定义.....	23
2.3.2 函数调用.....	27
2.3.3 JavaScript 函数的参数隐含调用.....	31
2.4 JavaScript 程序中的数组使用.....	33
2.4.1 数组的定义.....	33
2.4.2 数组的使用.....	35
2.4.3 创建数组的一个方法.....	37
2.4.4 内部数组.....	38
2.5 范例.....	39
2.5.1 Sample2_1: 用 JavaScript 做进度条.....	39
2.5.2 Sample2_2: 颜色变化的例子.....	40
第 3 章 JavaScript 定义的常用对象.....	43
3.1 基于对象的 JavaScript 语言.....	44
3.1.1 对象的基本结构.....	44

---

3.1.2	对象操作语句 .....	44
3.1.3	对象的引用途径 .....	46
3.1.4	对象属性的引用 .....	46
3.1.5	对象方法的引用 .....	46
3.2	新对象的创建 .....	48
3.2.1	对象的定义 .....	48
3.2.2	创建对象实例 .....	49
3.2.3	对象方法的使用 .....	49
3.3	JavaScript 中常用对象和函数 .....	51
3.3.1	JavaScript 中的系统函数 .....	51
3.3.2	常用内部对象 .....	51
3.4	文字列对象 .....	52
3.4.1	文字列对象的属性 .....	52
3.4.2	文字列对象的方法 .....	52
3.5	Math 对象 .....	61
3.5.1	主要属性 .....	61
3.5.2	主要方法 .....	61
3.6	Date 对象和 Timer 处理 .....	63
3.7	document 对象 .....	66
3.8	Navigator 对象 .....	69
3.9	history 对象、location 对象和 links 对象配列 .....	71
3.10	forms 配列和 elements 配列 .....	75
3.11	范例 .....	80
3.11.1	Sample3_1: 一个时钟显示的 JavaScript 文档 .....	80
3.11.2	Sample3_2: 停留时间 .....	81
3.11.3	Sample3_3: 主人的问候 .....	83
<b>第 4 章</b>	<b>表单的使用和 Event 处理 .....</b>	<b>85</b>
4.1	JavaScript 中的 Form 表单对象 .....	86
4.1.1	什么是表单对象 .....	86
4.1.2	Form 表单对象的方法 .....	87
4.1.3	Form 表单对象的属性 .....	87
4.1.4	访问表单对象 .....	87
4.1.5	this 和 this.form .....	88
4.2	Event 处理 .....	91
4.2.1	Event 种类 .....	91
4.2.2	Event 处理的实例: onFocus、onChange 及 onMouseOver .....	93
4.3	表单的基本元素 .....	96

---

4.3.1	表单元素的基本种类.....	96
4.3.2	应用举例: RadioButton、Select Box 和 Check Box .....	100
4.4	JavaScript 与 cookies 的使用 .....	106
4.4.1	什么是 cookies .....	106
4.4.2	如何判断客户端浏览器是否支持 JavaScript 和 cookies .....	109
4.5	反馈表单: Form 对象的发送 .....	110
4.5.1	反馈表单 .....	110
4.5.2	反馈表单中控制文字域的值.....	111
4.5.3	文字域关联事件 .....	112
4.5.4	反馈表单的预处理 .....	113
4.5.5	Form 内容的确认和发送.....	116
4.5.6	向多个服务器 CGI 发送同一表单.....	119
4.6	CGI 和服务器端反馈表单处理.....	120
4.6.1	HTML Form 格式部分.....	121
4.6.2	查询 CGI 程序.....	122
4.7	范例.....	125
4.7.1	Sample4_1.htm: 调用动态按钮文档 .....	125
4.7.2	Sample4_2.htm: 动态按钮程序 .....	125
<b>第 5 章</b>	<b>window 和 Frame 的管理.....</b>	<b>129</b>
5.1	window 对象的管理.....	130
5.1.1	如何打开一个新窗口.....	130
5.1.2	window 特性.....	131
5.1.3	window 间的参照.....	133
5.1.4	屏幕 (screen) 对象属性 .....	136
5.2	Windows 对象应用: 弹出窗口技巧 (1) .....	137
5.2.1	最基本的弹出窗口代码.....	137
5.2.2	经过设置后的弹出窗口.....	138
5.2.3	用函数控制弹出窗口.....	139
5.2.4	弹出窗口的定时关闭控制.....	140
5.2.5	弹出的窗口的 cookie 控制.....	141
5.3	Windows 对象应用: 弹出窗口技巧 (2) .....	142
5.3.1	用 SUBMIT 弹出窗口 .....	143
5.3.2	用 target 属性更换窗口内容.....	143
5.3.3	弹出图片窗口 .....	144
5.3.4	弹出窗口的生成器 .....	144
5.3.5	弹出窗口的 Focus .....	145
5.3.6	定位窗口 .....	146



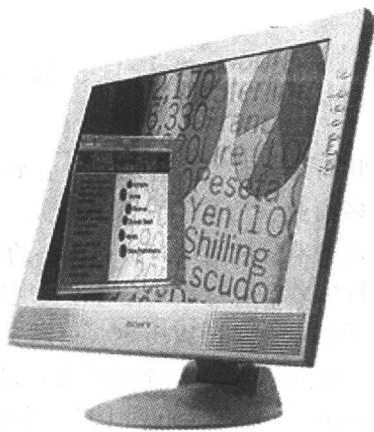
5.3.7	页面自动转换 .....	146
5.3.8	页面自动更新 .....	147
5.4	输入输出对话框 .....	147
5.4.1	输入对话框 .....	147
5.4.2	输出对话框 .....	148
5.5	Frame 框架对象：实现复杂的交互设计 .....	149
5.5.1	什么是框架 .....	149
5.5.2	框架的树状结构 .....	150
5.5.3	Frame 间的参照 .....	151
5.5.4	实例 .....	151
5.6	如何制作浮动层 .....	154
5.6.1	CSS .....	154
5.6.2	制作一个可以移动的静止浮动层 .....	156
5.6.3	制作一个漫游的自由浮动层 .....	160
5.7	在网页中加入菜单 .....	163
5.7.1	最简单的菜单条 .....	164
5.7.2	采用表格的方式设计菜单条 .....	165
5.7.3	下拉菜单设计 .....	166
5.7.4	树状目录 .....	167
5.8	范例 .....	173
5.8.1	Sample5_1：一种动态下拉菜单 .....	173
5.8.2	Sample5_2：垂直滚动看板 .....	176
<b>第 6 章</b>	<b>JavaScript 和图像处理 .....</b>	<b>179</b>
6.1	Image 对象 .....	180
6.1.1	Web 页上的图像 .....	180
6.1.2	图像数组 .....	182
6.2	图片的预先读取 .....	183
6.3	用 JavaScript 实现动画 .....	186
6.3.1	图像切换 .....	186
6.3.2	实现动画 .....	188
6.4	用 JavaScript 实现图像特效 .....	189
6.5	范例 .....	192
6.5.1	图形淡出淡隐 .....	192
6.5.2	雪花飘飘的背景 .....	194
<b>第 7 章</b>	<b>JavaScript 和 ActiveX 技术 .....</b>	<b>199</b>
7.1	ActiveX 组件 .....	200

---

7.1.1	什么是 ActiveX 组件 .....	200
7.1.2	建立 ActiveX DLL .....	200
7.1.3	ActiveX 组件与系统 .....	202
7.2	用户端 ActiveX 组件的应用 .....	203
7.2.1	网页中的 Flash 动画对象 .....	203
7.2.2	Flash 对象的方法 .....	204
7.2.3	JavaScript 对 Flash 对象的调用 .....	205
7.3	服务器端的 ASP 技术 .....	208
7.3.1	ASP 的发展 .....	208
7.3.2	ASP 网站设计环境 .....	209
7.3.3	ASP 的对象结构 .....	211
7.4	ASP 与 ActiveX 组件 .....	215
7.5	ASP 中的 JScript 脚本语言 .....	219
7.5.1	脚本语言 .....	219
7.5.2	设置脚本语言 .....	219
7.5.3	服务器端 JScript 脚本 .....	222
7.5.4	JScript 和 VBScript 的交互使用 .....	223
7.6	ASP 与表单处理 .....	225
7.7	Netscape 的 JavaScript 脚本语言 .....	228
7.7.1	client 对象 .....	228
7.7.2	request 对象 .....	230
7.7.3	server 对象 .....	231
7.7.4	File 对象 .....	231
7.7.5	callC()和 registerCFunction()函数 .....	232
<b>第 8 章</b>	<b>JavaScript 和数据库调用 .....</b>	<b>235</b>
8.1	数据库与网站构建 .....	236
8.1.1	构筑网络数据库 .....	236
8.1.2	构建一个 Linux 网站 .....	237
8.1.3	构筑一个 Windows NT 网站 .....	238
8.2	数据库查询语言 .....	238
8.2.1	SQL .....	238
8.2.2	ASP 中的 SQL 语言 .....	241
8.2.3	其他 SQL 语句 .....	243
8.3	数据库连接 .....	245
8.3.1	ODBC 的 DSN 方法连接 .....	246
8.3.2	ODBC 的非 DSN 方法连接 .....	249
8.3.3	OLE DB 的连接方法 .....	251

---

8.4 用 ASP 查询数据库 .....	252
8.4.1 用 Connection 对象查询数据库 .....	252
8.4.2 使用 Recordset 对象查询数据库 .....	253
8.4.3 结合 Form 表单的反馈信息访问数据库 .....	256
8.5 JavaScript 脚本访问数据库 .....	257
<b>第 9 章 实战 JavaScript</b> .....	<b>261</b>
9.1 鼠标特效制作 .....	262
9.1.1 鼠标特效——跟随鼠标的文字列 .....	262
9.1.2 鼠标特效——跟随鼠标的动画显示 .....	265
9.2 多媒体网页制作 .....	268
9.2.1 声频播放 .....	268
9.2.2 视频播放 .....	274
9.3 JavaScript 在 E-mail 中的应用 .....	276
9.3.1 在 E-mail 中使用 JavaScript 的可行性 .....	276
9.3.2 在弹出的对话框内显示邮件内容 .....	276
9.3.3 自动在新的窗口内显示邮件内容 .....	277
9.3.4 动态控制 E-mail 背景的颜色 .....	278
9.3.5 根据收信日期显示不同的邮件内容 .....	279
<b>参考文献</b> .....	<b>281</b>



# 第 1 章

## JavaScript 概要

---

本章主要介绍 JavaScript 语言在互联网技术中的应用背景、语言特征、脚本的基本书写方法以及 JavaScript 技术的相关资料，这些都是以后几章学习的基础。

## 1.1 JavaScript 语言概况

互联网带给现代社会全新的工作和生活方式，其互联性、开放性和共享信息的模式带来了许多商业、信息和技术的飞跃。那么怎样把需要开放的信息资源和服务内容加入到 WWW 服务器是广大用户日益关心的问题。采用超链接技术（超文本和超媒体技术）是实现这个目标最常用的方法。具体实现这种手段的支持环境，就是 HTML 超文本标识语言，利用它可制作所需的 Web 网页。超文本（Hyper Text）和超媒体（Hyper Media）技术结合超链接（Hyper Link）的链接功能可以将各种信息组织成网络结构（Web），构成网络文档（Document），实现互联网上的“漫游”，通过 HTML 符号的描述就可以实现文字、表格、声音、图像、动画等多媒体信息的检索。

但是，HTML 提供的这种超链接技术存在一定的缺陷，那就是它只能提供一种静态的信息资源，缺少动态的客户端与服务器端的交互。虽然可通过 CGI（Common Gateway Interface）通用网关接口实现一定的交互，但由于该方法编程较为复杂，因而在一段时间妨碍了 Internet 技术的发展。而 JavaScript 的出现，使得信息和用户之间不仅只是一种显示和浏览的关系，还是一种实时的、动态的、可交互的关系。因此基于 CGI 静态的 HTML 页面将被可提供动态实时信息、并对客户操作进行反应的 Web 页面取代。JavaScript 脚本正是满足这种需求而产生的。它深受广大用户的喜爱和欢迎，是众多脚本语言中较为优秀的一种。

JavaScript 由 Netscape 公司开发，是一种基于对象（Object）和事件驱动（Event Driven）并具有安全性能的脚本语言。使用它的目的是与 HTML 超文本标记语言、Java 脚本语言（Java 小程序）一起实现在一个 Web 页面中链接多个对象，与 Web 客户实现交互。它是通过在标准的 HTML 语言中嵌入或调入方式实现的，它的出现弥补了 HTML 语言的缺陷，它是 Java 与 HTML 折衷的选择，具有以下几个基本特点：

- ◆ JavaScript 是一种脚本语言。它采用小程序段的方式实现编程。像其他脚本语言一样，JavaScript 同样也是一种解释性语言，它提供了一个简易的开发过程。它的基本结构形式与 C、C++、VB、Delphi 十分类似。但它不像这些语言一样，需要先编译，而是在程序运行过程中被逐行地解释。它与 HTML 标识结合在一起，从而方便用户的使用和操作。
- ◆ JavaScript 是一种基于对象的语言。这意味着它能运用自己已经创建的对象。因此，许多功能来自于脚本环境中对象的方法和属性的调用。
- ◆ JavaScript 的主要特征是实现网页的动态化。它可以直接对用户或客户的输入作出响应，无须经过 Web 服务程序。它对用户的响应，是采用以事件驱动的方式进行的。所谓事件驱动，就是指在主页（Home Page）中执行了某种操作所产生的动作，就称为“事件”（Event）。比如按下鼠标、移动窗口、选择菜单等都可以视为事件。当事件发生后，可能会引起相应的事件响应。
- ◆ JavaScript 具有安全性。它不允许访问本地硬盘，不能将数据存入到服务器上，不

允许对网络文档进行修改和删除，只能通过浏览器实现信息浏览或动态交互，从而有效地防止数据丢失。

- ◆ JavaScript 具有简单性。首先，它是一种基于 Java 基本语句和控制流之上的简化语言，对于学习 Java 是非常好的过渡；其次，它的变量类型是采用弱类型，并未使用严格的数据类型。
- ◆ JavaScript 是依赖于浏览器中的 JavaScript 解释器来运行，与操作环境无关，只要计算机上装有支持 JavaScript 的浏览器就可正确执行，从而实现了“编写一次，走遍天下”的梦想。

实际上，JavaScript 最杰出之处在于可以用很小的程序做很多的事情。不一定必须得有高性能的电脑，软件仅需一个字处理软件及一个浏览器，无需 Web 服务器通道，通过自己的电脑即可完成所有的事情。综上所述，JavaScript 是一种新的描述语言，它可以嵌入到 HTML 的文件之中。JavaScript 语言可以在客户端回应使用者的需求事件（如：Form 的输入），而不需任何网络来回传输资料，所以当一位使用者输入一项资料时，它不需经过传给服务器端（Server）处理再传回来的过程，而直接可以被客户端（Client）的应用程序处理。

## 1.2 JavaScript 与 Java 的比较

HTML 就是 Hyper Text Markup Language 的缩写，它是网页浏览器采用的网页文本编辑语言，是执行文章格式控制的语言。其缺点是不能像一般程序语言那样实现程序控制流程。但是 HTML 难以实现的这些功能可以通过使用 Java 或 JavaScript 得到弥补。

JavaScript 在编程风格上主要具有如下特征：

- ◆ 引入 for 和 if 等控制语句使程序效率化；
- ◆ 目标模块（Document、Math、Date、文字列模块等）的导入使程序模块化；
- ◆ 网页、Frame 及 Window 的管理增强了网页设计的技巧；
- ◆ OnClick、OnChange 等事件处理的实现使网页可以在客户端作动态响应；
- ◆ Timer 处理的实现可以实现动态网页；
- ◆ 对网页端的数据输入输出对话框的实现；
- ◆ Image 目标模块对应的图像动态操作。

JavaScript 的语言格式与 C 语言很相似，有 C 语言编程经验的人能很容易掌握 JavaScript 语言。

JavaScript 虽然和 Java 都有同样的“Java”一词在内，但其实它们有很大差别。它们是两个公司开发的不同产品。Java 是 Sun 公司推出的新一代面向对象的程序设计语言，最初是为机顶盒设计的，后来发展起来并为互联网所应用，它的前身是 Oak 语言；而 JavaScript 是 Netscape 公司的产品，其目的是为了扩展 Netscape Navigator 功能而开发的一种可以嵌入 Web 页面中的基于对象和事件驱动的解释性语言，它的前身是 Live Script。两种语言在实际程序设计风格上还有以下差异：

#### ◆ 基于对象和面向对象

Java 是一种真正的面向对象的程序语言，还可以自行设计各种需要的对象模块。JavaScript 是一种脚本语言，是一种基于对象（Object Based）和事件驱动（Event Driver）的编程语言，因而它本身提供了非常丰富的内部对象供设计人员使用。

#### ◆ 解释和编译

两种语言在其浏览器中所执行的方式不一样。Java 的源代码在传递到客户端执行之前，必须经过编译，因而客户端上必须具有相应平台上的仿真器或解释器，它可以通过编译器或解释器实现独立于某个特定的平台编译代码。

JavaScript 是一种解释性编程语言，其源代码在发往客户端执行之前不需经过编译，而是将文本格式的字符代码发送给客户端，由浏览器解释执行。

#### ◆ 强变量和弱变量

两种语言所采取的变量是不一样的。Java 采用强类型变量检查，即所有变量在编译之前必须作声明。如：

```
Integer x;  
String y;  
x=100;  
y="100";
```

其中 x 是一个整数，y 是一个字符串。

JavaScript 中变量声明采用其弱类型。即变量在使用前不需作声明，而是解释器在运行时检查其数据类型。如：

```
x=100;  
y="100";
```

前者说明 x 为其数值型变量，而后者说明 y 为字符型变量。

#### ◆ 代码格式不一样

Java 是一种与 HTML 无关的格式，其代码以字节代码的形式保存在独立的文档中。JavaScript 的代码是一种文本字符格式，可以直接嵌入到 HTML 文档中，并且可动态装载。编写 HTML 文档就像编辑文本文件一样方便。

#### ◆ 嵌入方式不一样

在 HTML 文档中，两种编程语言的标识不同，JavaScript 使用<Script>...</Script>来标识，而 Java 使用<applet>...</applet>来标识。

#### ◆ 静态联编和动态联编

Java 采用静态联编，即 Java 的对象引用必须在编译时进行，以使编译器能够实现强类型检查。JavaScript 采用动态联编，即 JavaScript 的对象引用在运行时进行检查，如不经编译就无法实现对象引用的检查。

尽管有这么多的差异，但 JavaScript 和 JavaApplet 都紧密地与 HTML 结合在一起，使

网页实现多种动态功能。无论 JavaScript 还是 JavaApplet，使用时都需要插入到 HTML，通过浏览器载入执行。

由于 JavaApplet 装载的时间较长，如果网页中的 JavaApplet 程序太多会导致页面下载很慢，这也许使得访问者没有太多的耐心等待页面完全显示出来就会离开了。JavaScript 可以看作是 Java 的“亲戚”，它实际上是一种脚本语言，具有文本性质，所以我们能够看到它的源代码，非常便于学习和理解，稍有编程经验的人都能迅速改变和修改其源代码以适应自己的页面，是一种真正意义上的面向普通用户的脚本语言。JavaScript 脚本代码一般插入到 HTML 的<head>或<body>区，当浏览器浏览有 JavaScript 脚本的页面时就会执行并产生“交互”的动态页面效果，速度非常快。现在凡是制作得比较出色的网站无一例外都或多或少地使用了 JavaScript 技术。

1

## 1.3 HTML 中的 JavaScript 脚本

下面是一个编写 JavaScript 脚本的简单例子：

```
<SCRIPT LANGUAGE="JavaScript">
document.write("<FONT COLOR='RED'>红色文字</FONT>")
</SCRIPT>
```

需要注意的是，JavaScript 脚本必须要浏览器支持才可以执行。好在，应用广泛的 Netscape 和 Internet Explorer 现在都支持 JavaScript。应该说 JavaScript 脚本有广泛的应用空间。

如何书写 JavaScript 呢？

首先，JavaScript 不是 HTML，而是 HTML 中的一段文本。HTML 的语法并不是非常严谨，而是比较宽松的。比如在文字间或者段落间加入若干的空格对于 HTML 并没有多大的区别，这也就是为什么大多数 HTML 语言没有必要非得写在同一行，写不写在一行这对于 HTML 没有任何的区别。JavaScript 的语法规则就要严格得多。如果这样写上面的例子：

```
document.write("<H1>早上好! </H1>")
```

因为没有分号“;”就断行了，执行时就会出错（出错提示为：无效的结尾）。

另一个问题是，JavaScript 对大小写也是有区分的。JavaScript 的首行应该是：

```
<SCRIPT LANGUAGE="JavaScript">
```

这一句告诉浏览器“下面开始执行 JavaScript 脚本了”。因为还有许多其他类型的脚本存在，例如 LiveScript、VBscript 等，所以这一句里的 LANGUAGE= "JavaScript"是必需的，加上这一句使浏览器很轻松地理解文本的意思。JavaScript 脚本以<SCRIPT LANGUAGE="JavaScript">开始，以</SCRIPT>结束。浏览器在解释 HTML 时遇到<SCRIPT LANGUAGE="JavaScript"> ...</SCRIPT>语句，就把其间的语句交给 JavaScript 解释器去执行，然后再把 HTML 翻译为所浏览的网页。



可将 `<Script>...</Script>` 标识放入 `<Head>...</Head>` 或 `<Body>...</Body>` 之间。将 JavaScript 标识放置在 `<Head>...</Head>` 头部之间，使其在主页和其余部分代码之前装载，从而可使代码的功能更强大；可以将 JavaScript 标识放置在 `<Body>...</Body>` 主体之间以实现某些部分动态地创建文档。

再看看上面的例子（稍微解释一下）：

```
document.write("<H1>早上好! </H1>")
```

这一句中，`document` 是一个“已存在的对象”，实际指的是 HTML 文件的实体，`write` 是 `document` 对象所提供的“方法”，`document.write()` 是文档对象的输出函数，它的功能是把“HTML 文本”写到浏览器，最后直接编写到网页的 HTML 文件中。于是脚本就明白了调用对象的 `write` 方法要写什么东西，括弧里面就是要写的东西，可以使用标准的 HTML 语法。需要注意的是双引号里面要用单引号，要配对才行。否则脚本运行会出错。

下面编写一个 JavaScript 程序。JavaScript 程序的分隔符是分号“;”，这与 C 语言风格极其相近，如示例 1.1。

示例 1.1

```
<HTML>
<HEAD>
<TITLE>网页文本</TITLE>
</HEAD>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
  document.write(" <HR>");
  document.write("<H1>欢迎访问 IC 网络公司主页</H1>");
  document.write("<HR>");
</SCRIPT>
</BODY>
</HTML>
```

这个程序在 Netscape Navigator 和 Microsoft IE 上显示的结果如图 1.1 和图 1.2 所示。

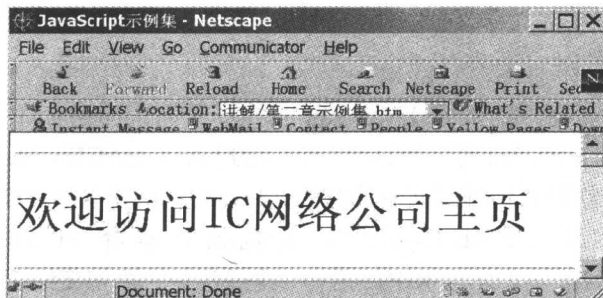


图 1.1 示例 1.1 在 Netscape Navigator 上的运行结果