



事半功倍系列丛书

JavaScript™

Kelly Murdock 著

侯或 刘昕 译

创建**动态**网页的**高级**脚本语言



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



事半功倍系列丛书

JavaScript

Kelly Murdock 著

侯或 刘昕 译

清华大学出版社

(京)新登字 158 号

内 容 简 介

在网络经济飞速发展的今天,创建富有吸引力的动态网页对个人和公司而言都具有重大的意义,而 JavaScript 正是您创建动态网页的最流行的语言。

本书是事半功倍系列丛书之一,全面地为您介绍 JavaScript 的各个方面,包括使用变量和数组、创建表达式、控制程序流程、使用函数、处理事件、使用对象、使用窗口和字符串、使用日期和时间、使用 MATH 对象、使用表单、分栏、决定环境属性以及调试 Javascript 等等。本书图文并茂、通俗易懂,将会是您创建动态网页的有力助手。

JavaScript

Copyright © 2001 by 清华大学出版社.

Original English language edition copyright © 2000 by IDG Books Worldwide, Inc.

All rights reserved including the right of reproduction in whole or in part in any form.

This edition published by arrangement with the original publisher, IDG Books Worldwide, Inc., Foster City, California, USA.

本书中文简体字版由 IDG 图书公司授权清华大学出版社出版,未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号:图字 01-2001-0119 号

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

作 者: Kelly Murdock

译 者: 候 彧 刘 昕

责任编辑: 夏兆彦

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>

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: 850×1168 1/16 印张: 19 字数: 446 千字

版 次: 2001 年 8 月第 1 版 2001 年 8 月第 1 次印刷

书 号: ISBN 7-302-04357-4/TP·2558

印 数: 0001~5000

定 价: 27.00 元(全套丛书共 7 册,总定价 201.50 元)

目 录

第 1 章 JavaScript 简介	1
1.1 什么是脚本编制.....	2
1.2 在 URL 栏输入 JavaScript 语句.....	4
1.3 将 JavaScript 嵌入 HTML 文档.....	6
1.4 链接到外部 JavaScript 文件	8
1.5 将内容呈递给非 JavaScript 浏览器.....	10
1.6 将注释添加到 JavaScript.....	12
第 2 章 使用变量和数组	15
2.1 了解变量类型.....	16
2.2 声明变量.....	18
2.3 给变量赋值.....	19
2.4 使用整数.....	20
2.5 使用浮点数.....	21
2.6 使用布尔值.....	22
2.7 使用字符串.....	23
2.8 确定变量类型.....	24
2.9 将字符串转换为数字.....	26
2.10 把数字转换为字符串.....	28
2.11 声明数组.....	30
2.12 确定数组的元素个数.....	32
2.13 将数组转换为字符串.....	34
2.14 对数组排序.....	36

第 3 章 创建表达式	39
3.1 使用算术运算符	40
3.2 递增变量和递减变量	42
3.3 创建比较表达式	44
3.4 创建逻辑表达式	46
3.5 使用条件运算符	48
3.6 了解优先顺序	50
3.7 字符串表达式的求值	51
3.8 识别数字	52
第 4 章 控制程序流程	55
4.1 使用 If-else 语句	56
4.2 使用 for 循环	58
4.3 使用 while 循环	60
4.4 中断循环	61
4.5 继续循环	63
4.6 使用 JavaScript 定时器	64
4.7 设置定期间隔	66
4.8 清除超时和间隔	68
第 5 章 使用函数	71
5.1 声明函数	72
5.2 调用函数	73
5.3 了解全局变量和局部变量	74
5.4 将参数传送给函数	76
5.5 从函数返回值	78
5.6 通过 HTML 链接调用函数	80
第 6 章 处理事件	83
6.1 使用 JavaScript 事件	84

6.2	检测鼠标单击.....	86
6.3	检测双击.....	88
6.4	创建悬停按钮.....	90
6.5	检测按键.....	92
6.6	设置焦点.....	94
6.7	检测下拉菜单选择.....	96
6.8	创建网页加载和卸载信息.....	98
第 7 章 使用对象.....		101
7.1	理解对象、属性和方法.....	102
7.2	使用网页元素对象.....	104
7.3	使用子对象.....	106
7.4	使用预定义对象.....	108
7.5	创建新的对象.....	110
7.6	引用当前对象.....	112
7.7	查看对象属性.....	114
7.8	使用 Array 对象.....	116
7.9	使用 image 对象.....	118
7.10	预加载图像.....	119
7.11	改变图像.....	120
7.12	使用 link 和 anchor 对象.....	122
7.13	改变链接.....	124
7.14	使用 History 对象.....	126
第 8 章 使用窗口.....		129
8.1	使用窗口和文档对象.....	130
8.2	在浏览器的状态栏上显示文本.....	131
8.3	改变背景色.....	134
8.4	改变文本和链接颜色.....	136
8.5	改变文档标题.....	138
8.6	显示修改日期.....	139
8.7	查看当前文档的 URL.....	140
8.8	查看引用页.....	141



8.9	打开新的浏览器窗口.....	142
8.10	关闭远程窗口.....	144
8.11	打印窗口.....	146
8.12	移动窗口.....	148
8.13	改变窗口大小.....	150
8.14	用警告对话框通知用户.....	152
8.15	用提示对话框接受输入.....	154
8.16	用确认对话框使用户做出决定.....	156
第9章 使用字符串.....		159
9.1	使用字符串对象.....	160
9.2	使用子字符串.....	162
9.3	连接字符串.....	164
9.4	格式化字符串变量.....	166
9.5	创建锚和链接.....	168
9.6	确定字符串长度.....	170
9.7	改变大小写.....	171
9.8	在字符串内搜索.....	172
9.9	定位字符串中的字符.....	174
9.10	替换字符串中的文本.....	176
9.11	分离字符串.....	178
第10章 使用日期和时间.....		181
10.1	使用 Date 对象.....	182
10.2	显示当地时间和日期.....	184
10.3	获得时间和日期值.....	186
10.4	设置时间和日期值.....	188
10.5	使用时间.....	190
10.6	使用时区.....	192
第11章 使用 Math 对象.....		195
11.1	使用 Math 对象.....	196



11.2	生成随机数.....	198
11.3	使用数学常数.....	200
11.4	使用三角函数.....	202
11.5	使用对数和指数.....	204
11.6	使用平方根.....	205
11.7	数字的舍入.....	206
11.8	对数字进行乘方运算.....	207
11.9	发现最小值和最大值.....	208
11.10	使用绝对值.....	209
 第 12 章 使用表单.....		 211
12.1	使用文本框.....	212
12.2	使用密码框.....	214
12.3	使用隐藏字段.....	215
12.4	使用文本区域框.....	216
12.5	使用按钮.....	218
12.6	使用重置按钮.....	220
12.7	使用【提交】按钮.....	221
12.8	使用复选框按钮.....	222
12.9	使用单选按钮.....	224
12.10	使用选择列表.....	226
12.11	使用单独的选择列表项.....	228
12.12	使用多项选择列表.....	230
12.13	停用表单元素.....	232
12.14	重置和提交表单.....	234
12.15	验证表单的有效性.....	236
12.16	控制表单焦点.....	238
12.17	使用焦点事件.....	240
 第 13 章 使用分栏.....		 243
13.1	使用 Frame 对象.....	244
13.2	根据名称访问分栏.....	246
13.3	发现分栏数.....	248



13.4	引用当前分栏.....	249
13.5	跨分栏的引用.....	250
13.6	向一个单独的分栏写入文本.....	252
13.7	检测分栏的大小调整.....	253
13.8	设置目标分栏.....	254
13.9	打印分栏.....	255
第 14 章	使用 navigator 对象	257
14.1	使用 navigator 对象	258
14.2	检测用户的浏览器.....	260
14.3	检测浏览器的版本.....	261
14.4	检测用户的操作系统.....	262
14.5	了解用户代理.....	263
14.6	确定是否启动了 Java	264
14.7	确定代码名称.....	265
14.8	使用 location 对象.....	266
14.9	使用 Cookie	268
14.10	将字符编码.....	270
第 15 章	检测 JavaScript 错误	273
15.1	检测 JavaScript 错误	274
15.2	使用注释进行调试.....	276
15.3	向文本框输出值.....	278
15.4	常见的 JavaScript 错误	280
附录 A	HTML4.01 参考	282



1

JavaScript 简介

- 什么是脚本编制
- 在 URL 栏输入 JavaScript 语句
- 将 JavaScript 嵌入 HTML 文档
- 链接到外部 JavaScript 文件
- 将内容呈递给非 JavaScript 浏览器
- 将注释添加到 JavaScript

1.1 什么是脚本编制

用超文本标记语言(HTML)编写的网页文件定义诸如文本、表和图像这样的元素在浏览器中查看时的表现方式。HTML 只负责处理静态元素。

JavaScript 是一种脚本编制语言,能与 HTML 共同用于网页文件,还可以增加网页的交互性。这种交互性使网页创建者可以动态地控制网页元素。用户与网页交互时,您也可以使用 JavaScript 改变网页元素。例如,您可以在一个表单的文本框中设置文本,当用户单击按钮时进行更新。

JavaScript 可以在浏览器中运行,无需另外的工具。

什么是脚本编制

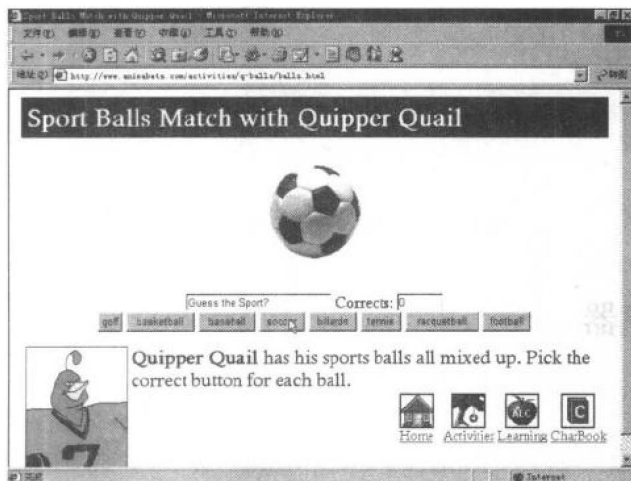


图 1.1

1 在浏览器中打开一个包括 JavaScript 的网页,如:
www.animabets.com/activities/qballs/balls.html.

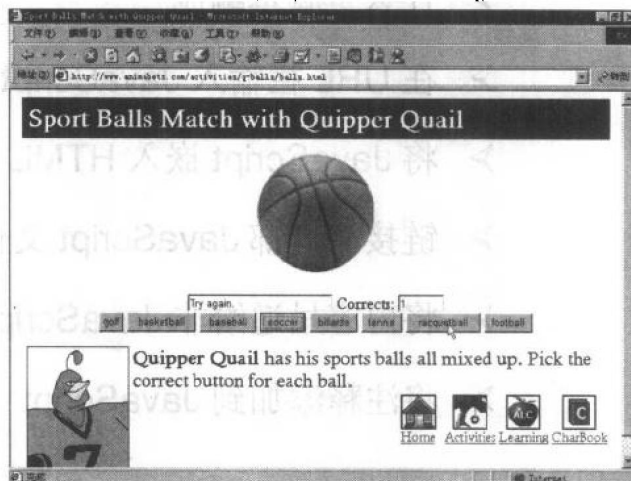


图 1.2

2 单击其中一个按钮。
3 单击另一个按钮。
文本框显示响应结果,显示出来一个新的图像。

注意: 这个网页的 HTML 文件包括定义文本框和图像如何更新的 JavaScript 语法。

补充说明

JavaScript 是最流行的基于客户机的 Web 脚本编制语言，因为 Netscape Navigator 和 Internet Explorer 都支持它，但是您可以使用其他脚本编制语言添加网页的交互性。将<script>标记的 language 属性设置为 VBScript，就可以将 VBScript 指定为脚本编制语言。VBScript 在语法上与 Visual Basic 类似，但只有 IE 浏览器支持它。VBScript 和 Visual Basic 编程语言很相似，如果您熟悉 Visual Basic 的话，使用 VBScript 的优势就在于它比 JavaScript 容易学。缺点是它只能在 IE 浏览器中使用。

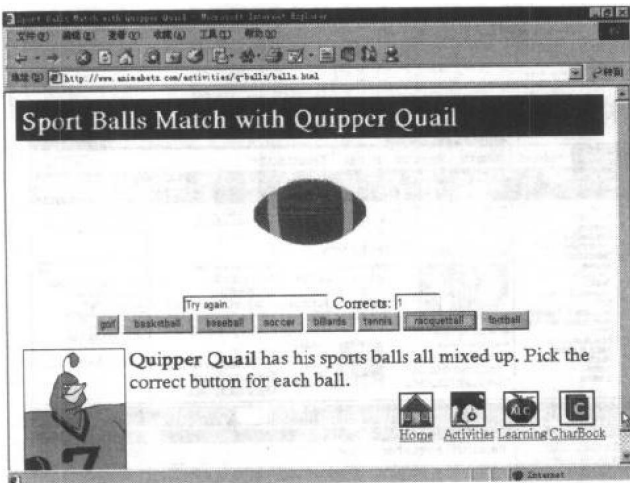


图 1.3

■ 文本框再次显示结果，又出现一幅新的图像。

4 选择【查看】|【源文件】查看该页的源代码。

注意：文本框视回答情况进行逻辑更新。



图 1.4

5 向下滚屏直到您能看到<script>标记。

■ 注意语言属性设置为 JavaScript。

1.2 在 URL 栏输入 JavaScript 语句

JavaScript 由一系列语句组成, 当执行时它们可以表现预期的功能。例如, JavaScript 语句 `document.write("hello")` 在浏览器中显示 *hello* 这个词。

浏览器能解释和执行 JavaScript 语句。这些语句可以在网页文件中找到, 也可以直接输入到浏览器的 URL 框中。

要让浏览器执行一个 JavaScript 语句, 请在浏览器的地址框中输入关键字 **javascript:**, 再输入要执行的 JavaScript 语句, 然后按 Enter 键就会执行该语句。

应该在 JavaScript 语句的末尾加上一个分号 (;)。地址框中可以包括几种不同的 JavaScript 语句, 中间用分号隔开。

例如, 如果您输入 `javascript:document.write("hello");` 当您按 Enter 键时浏览器会显示文本 “hello”。

在 URL 栏输入 JavaScript 语句

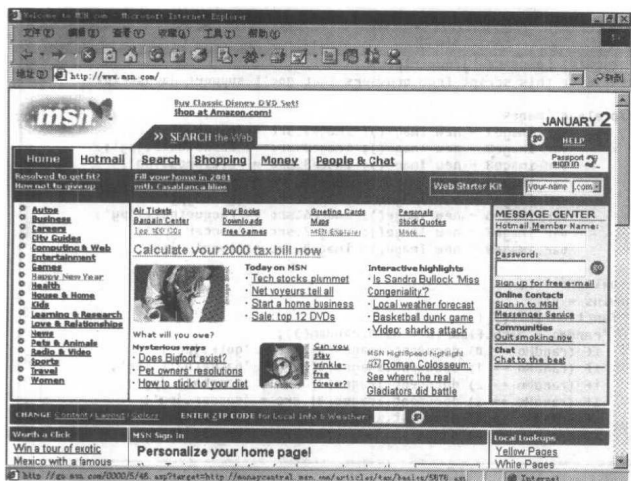


图 1.5
1 打开一个浏览器。

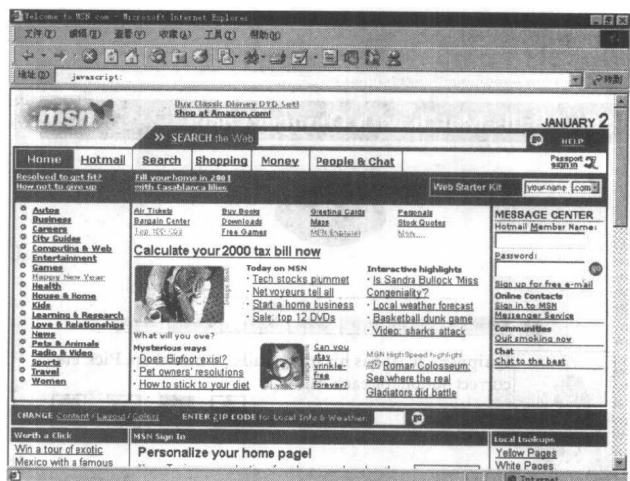


图 1.6
2 在地址框中, 输入 javascript: 关键字。

补充说明

如果在地址框中输入了一个关键字，浏览器一般会搜索基于此关键字的网页。如果您想让浏览器搜索 JavaScript 网页，而不是执行一个 JavaScript 语句，只需省略 javascript: 后面的冒号即可。如果您在浏览器的地址框中输入关键字 javascript: 时不包括冒号 (:), 浏览器就会搜索与关键字 javascript 匹配的网页。冒号的作用是告诉浏览器后面的文本应该解释为 JavaScript 语句。

如果您忘了输入关键字 javascript: , 浏览器会认为您输入的 JavaScript 语句是个网页地址。它将按那个地址去寻找网页，而这个网页它当然是找不到了。

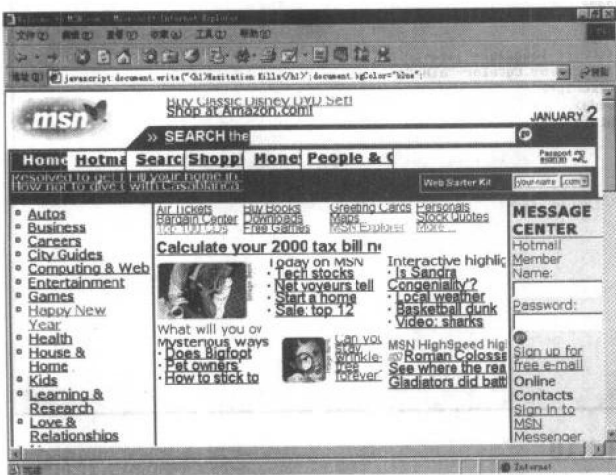


图 1.7

3 在 javascript:关键字后，输入 document.write("<h1>Hesitation Kills</h1>");。

4 再输入一个语句 document.bgColor="blue";。

5 按下 Enter 键。

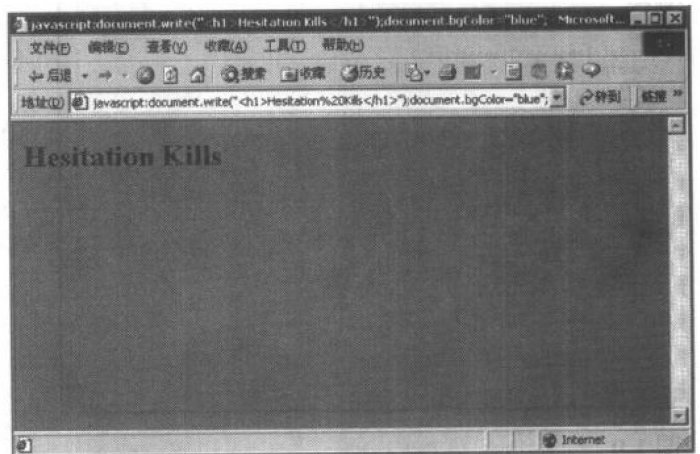


图 1.8

JavaScript 语句在浏览器中执行。

1.3 将 JavaScript 嵌入 HTML 文档

JavaScript 可以包括在网页的<script>标记内。当在浏览器中加载网页时执行包括在<script>标记内的 JavaScript 语句。

<script>标记可以出现在<head>或<body>标记内的任何一处。一个网页可以包括多套<script>标记。

<script>标记接受 type 和 language 属性。type 属性可以设置为被认可的 MIME 类型，如 text/javascript。language 属性是指定要使用的脚本编制语言的另一种方法。要使用 JavaScript 语言，只需把该属性设为 javascript 即可。

如果 JavaScript 语句在浏览器中显示任何文本，则该文本的位置就是网页文件中<script>标记所处的位置。例如，如果一个 HTML 文件有一个标题，后面跟着一个输出某些文本的<script>标记，那么该 JavaScript 文本就会紧接着 HTML 标题之后出现。

此例使用了本书光盘中的文件 skeleton.html，但这些步骤可以用于任何文件。

HTML 文档中嵌入 JavaScript

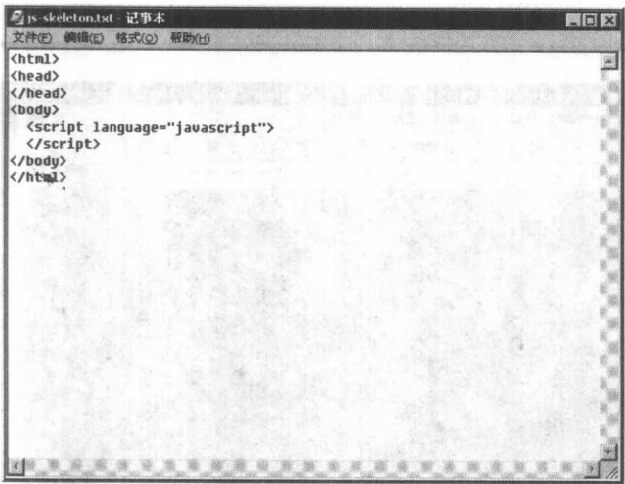


图 1.9

- 1 打开 skeleton.html 文件。
- 2 在<body>标记内添加一套<script>标记。
- 3 把语言属性添加到<script>标记。

- 4 设置语言属性等于 javascript。
- 5 把文件另存为 js-skeleton.html。

注意：该页能够用来轻松地创建一个包括 JavaScript 的网页。

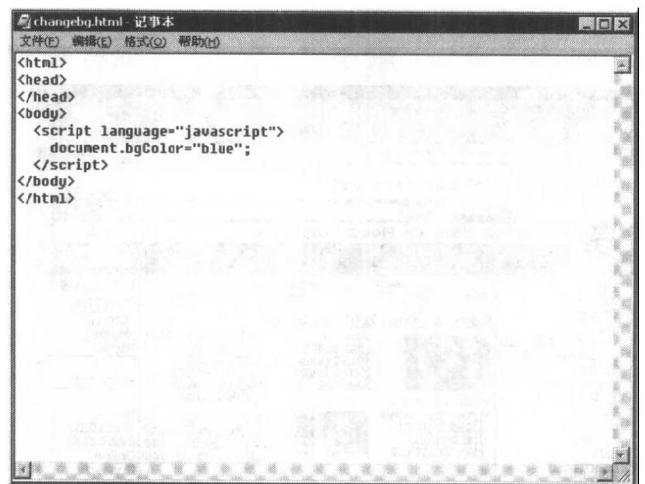


图 1.10

- 6 在<script>标记内，添加一个 JavaScript 语句改变背景色。
- 7 在 JavaScript 语句的末尾加上一个分号 (;)。

- 8 把文件另存为 changebg.html。注意：放在<script>标记内的 JavaScript 语句在加载该文件时执行。

学以致用

使用 JavaScript 您可以为访问者提供个性化网页。下面的代码片段能打开一个对话框，用户可以在里面输入名字。然后这个名字通过使用另一个 JavaScript 语句显示在网页中：

```
temp = window.prompt ("What is your name");  
document.write ("Welcome, " + temp + " to this web page. I hope  
you enjoy your visit. ");
```

要使此脚本发挥作用，您只需把它包括在一套<script>标记内，然后在浏览器中打开该文件即可。

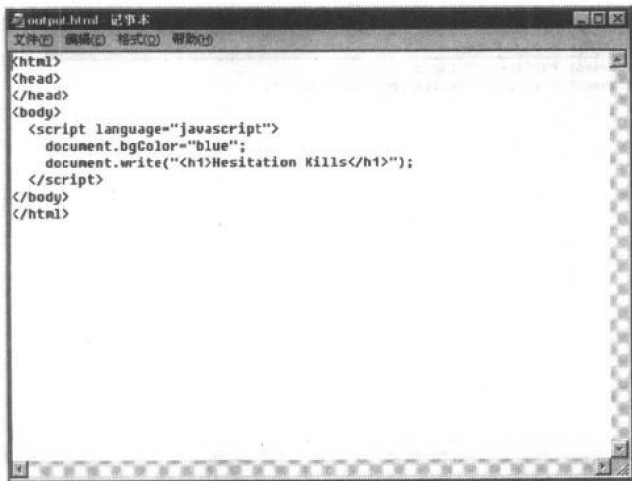


图 1.11

9 添加另一个 JavaScript 语句，输出一个短语。

■ 输出语句可以包括 HTML 标记。

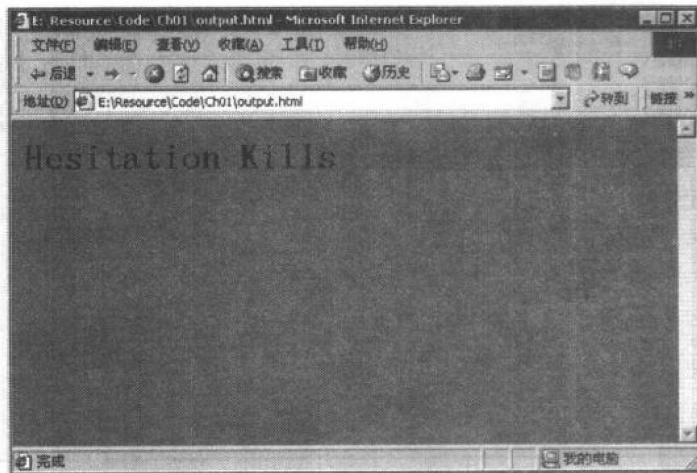


图 1.12

12 在浏览器中打开 output.html 文件。

■ 注意背景色如何改变，文本如何显示。

10 在 JavaScript 语句末尾加上一个分号(;)。

11 把文件另存为 output.html。

1.4 链接到外部 JavaScript 文件

另一个可以在<script>标记中使用的属性是 src 属性。该属性能够设置到包含 JavaScript 脚本的外部文件的 URL 中。

该外部文件不需包括<script>标记，只要包括 JavaScript 语句即可。它也不需包括 html 扩展名。外部 JavaScript 文件一般都有一个 js 的扩展名。

外部 JavaScript 文件可以在 Web 上的任何地方，可以通过后面加上该外部文件的 Web 地址的 http://协议关键字来引用。

例如，语句<script src="myfile.js">将会访问和该 HTML 文件在同一个目录中的、名为 myfile.js 的外部 JavaScript 文件。

此例使用了本书光盘中的文件 output.html (或您在“将 JavaScript 嵌入 HTML 文档”中创建的文件)，但您可以将这些步骤用于任何文件。

图 1.13 链接到外部 JavaScript 文件

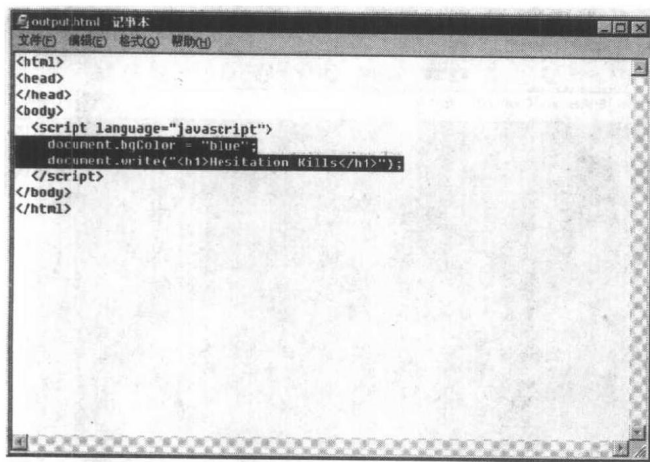


图 1.13

1 在【记事本】中打开 output.html 文件。

2 把<script>标记内的内容从文件中剪切出来。



3 把文件另存为 external.html。

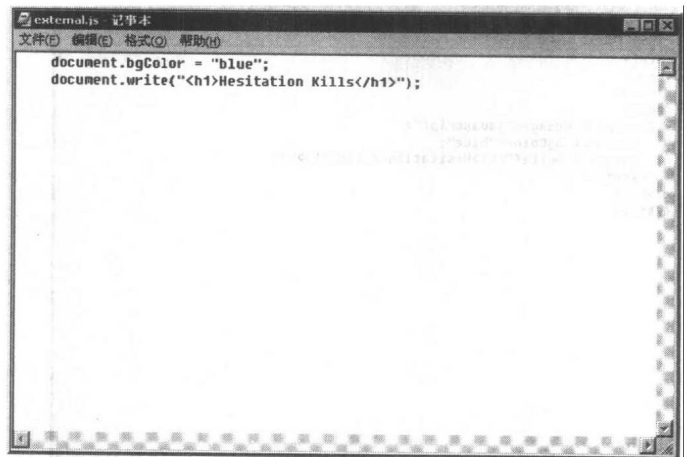


图 1.14

创建一个要链接的文件

1 在【记事本】打开一个新文件。

2 把从 output.html 文件<script>标记内剪切出来的内容粘贴进这个新文件。

3 把文件另存为 external.js。