



课堂实录 含CD-ROM

JavaScript 网页设计与 网站特效 课堂实录

刘贵国 晁代远 / 编著

- 配备光盘：
包含了大量相关素材及85个视频教学文件，使学习更加轻松、方便。
- 实用性强：
专业设计师及教学专家倾力奉献，案例全部来源于工作一线与教学实践。
- 案例专业：
包括导航菜单特效、文字和图片特效、
按钮链接和页面特效、广告代码特效。
- 体系完善：
涵盖JavaScript的基本知识、高级技巧、HTML基础、
window对象、Ajax基础等。



清华大学出版社



课堂实录

JavaScript 网页设计与 网站特效 课堂实录

刘贵国 晁代远 / 编著



清华大学出版社
北京

内容简介

本书全面介绍JavaScript的基本知识、高级技巧和实例,全书共17章,包括JavaScript基础知识、HTML基础、数据类型和变量、表达式与运算符、JavaScript程序核心语法、JavaScript核心对象、JavaScript中的事件、window对象、屏幕和浏览器对象、文档对象、历史对象和地址对象、表单对象和图片对象、Ajax基础、导航菜单特效案例、文字和图片特效案例、按钮链接和页面特效案例、广告代码特效案例等内容。

本书适合所有想全面学习JavaScript的开发人员阅读,包括网页设计与制作人员、JavaScript初学者、JavaScript专业开发人员、JavaScript开发爱好者、大中专院校的学生以及社会培训班学员。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

JavaScript网页设计与网站特效课堂实录 / 刘贵国 晁代远 编著. -- 北京:清华大学出版社, 2015
(课堂实录)

ISBN 978-7-302-39556-0

I. ①J… II. ①刘… ②晁… III. ①JAVA语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第046810号

责任编辑:陈绿春

封面设计:潘国文

责任校对:胡伟民

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:188mm×260mm

印 张:18.5

字 数:533千字

(附光盘1张)

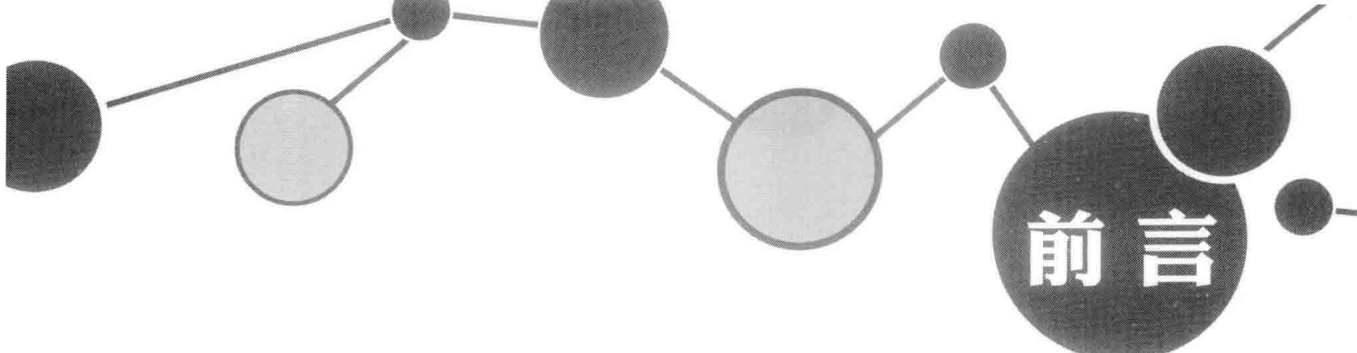
版 次:2015年10月第1版

印 次:2015年10月第1次印刷

印 数:1~3500

定 价:49.00元

产品编号:061948-01



前言

早些年, JavaScript由于其复杂的文档对象模型(DOM)和不一致的浏览器实现而不受开发者的待见。而随着技术的发展, JavaScript变得越来越强大、完善, 比如, Ajax技术可以创建更加迷人的Web应用, Node.js将JavaScript的应用范围扩展到了服务器端, 各种层出不穷的框架使得JavaScript的开发更加简捷, 尤其是近几年HTML5的出现, 将JavaScript提升到了前所未有的高度。如今JavaScript已经变成了Web开发必备的语言, 甚至开始逐步向移动领域渗透, 由于JavaScript的跨平台特性, 使得它在移动互联网时代可以有更大的作为。

JavaScript是面向Web的编程语言。绝大多数现代网站都使用了JavaScript, 并且所有的现代Web浏览器均包含了JavaScript解释器。这使得JavaScript成为使用最广泛的编程语言之一。JavaScript也是前端开发工程师必须掌握的3种技能之一: 描述网页内容的HTML、描述网页样式的CSS以及描述网页行为的JavaScript。本书能帮助你掌握JavaScript这门语言。

本书主要内容

JavaScript是目前网页设计中最简单易学并且易用的语言, 它能让你的网页更加生动活泼。利用JavaScript做出的网页特效, 能大大提高网页的可观性, 增加收藏和点击率。

本书共17章, 包括JavaScript基础知识、HTML基础、数据类型和变量、表达式与运算符、JavaScript程序核心语法、JavaScript核心对象、JavaScript中的事件、window对象、屏幕和浏览器对象、文档对象、历史对象和地址对象、表单对象和图片对象、Ajax基础、导航菜单特效案例、文字和图片特效案例、按钮链接和页面特效案例、广告代码特效案例等内容。

本书主要特色

● 知识全面系统

本书内容完全从网页创建的实际角度出发, 内容涉及JavaScript的基本知识、高级技巧和核心原理, 基本涵盖了JavaScript开发的所有重要知识和特效实例, 而且还介绍了结合jQuery的实例、JavaScript和CSS结合特效实例及Active技术应用等。

● 典型实例讲解

本书的每章都配有大量实用案例, 这些实例都来自于实际的网页开发实践, 实用性非常强, 读者通过研读这些实例, 可以系统地掌握JavaScript的开发技术。

● 配合Dreamweaver进行讲解

本书以浅显的语言和详细的步骤介绍了在可视化网页软件Dreamweaver中, 如何运用



JavaScript代码来创建网页，使网页制作更加得心应手。

● 配图丰富，效果直观

对于每一个实例代码，本书都配有相应的效果图，读者无须自己运行编码，也可以看到相应的运行结果或者显示效果。在不便上机操作的情况下，读者也可以根据书中的实例和效果图进行分析和比较。

● 贯穿大量的开发技巧，迅速提升开发水平

本书在讲解实例时贯穿了大量的网页开发技巧，通过对这些技巧的掌握，可以让读者掌握实际编程的捷径，从而迅速提高开发水平。

● 习题强化

每章后都附有针对性的练习题，通过实训巩固每章所学的知识。

本书读者对象

网页设计与制作人员。

网站建设与开发人员。

JavaScript初学者。

想全面学习JavaScript开发技术的人员。

JavaScript专业开发人员。

JavaScript开发爱好者。

大中专院校的学生。

社会培训班学员。

参加本书编写的人员包括：刘贵国、晁代远、张连元、晁辉、陈石送、何琛、吴秀红、王冬霞、何本军、乔海丽、邓仰伟、孙雷杰、孙文记、何立、倪庆军、胡秀娥、赵良涛、徐曦、刘桂香、葛俊科、葛俊彬等。由于时间所限，书中疏漏之处在所难免，恳请广大读者朋友批评指正。

作 者

目录

第1章 JavaScript基础知识

1.1 JavaScript简介.....2	1.3 第一个JavaScript程序.....6
1.1.1 JavaScript的历史.....2	1.3.1 预备知识.....6
1.1.2 JavaScript特点.....2	1.3.2 JavaScript编辑器的选择.....6
1.1.3 JavaScript和Java的区别.....3	1.3.3 编写Hello World程序.....6
1.2 在HTML中嵌入JavaScript的方法.....4	1.3.4 浏览器对JavaScript的支持.....7
1.2.1 <script/>使用方法.....4	1.4 实战应用——浏览器状态栏显示信息...8
1.2.2 外调脚本文件.....5	1.5 课后练习.....9
1.2.3 直接位于事件处理部分的代码中.....5	

第2章 HTML基础

2.1 HTML语言概述.....11	2.6 HTML图像标签.....23
2.1.1 HTML概述.....11	2.6.1 课堂小实例——属性img.....23
2.1.2 HTML文档结构.....12	2.6.2 课堂小实例——属性alt.....24
2.2 HTML标签.....13	2.6.3 课堂小实例——属性height和width.....25
2.2.1 课堂小实例——头部标签 <head>...</head>.....13	2.6.4 课堂小实例——属性align.....26
2.2.2 课堂小实例——标题标签 <title>...</title>.....13	2.6.5 课堂小实例——属性border.....27
2.2.3 课堂小实例——索引标签<isindex>...14	2.7 HTML表格标签.....28
2.2.4 主体标签<body>...</body>.....14	2.7.1 课堂小实例——表格标签 <table>...</table>.....28
2.2.5 课堂小实例——特殊字符.....15	2.7.2 课堂小实例——表头标签<th>...</th>.....28
2.3 HTML格式标签.....15	2.7.3 课堂小实例——说明标签 <caption>...</caption>.....29
2.3.1 课堂小实例——段落标签<p>.....15	2.8 HTML框架标签.....30
2.3.2 课堂小实例——换行标签16	2.8.1 框架集标签<frameset>...</frameset>...30
2.3.3 课堂小实例——缩进标签<blockquote>.....16	2.8.2 标签<noframes>.....31
2.3.4 课堂小实例——预格式化标签<pre>.....17	2.8.3 浮动窗口标签<iframe>...</iframe>...32
2.4 HTML文本标签.....18	2.8.4 框架的frameborder设置.....32
2.4.1 课堂小实例——标题标签.....18	2.9 HTML表单标签.....33
2.4.2 课堂小实例——标签.....19	2.9.1 课堂小实例——表单标签 <form>...</from>.....33
2.4.3 课堂小实例——字体大小标签font-size...19	2.9.2 课堂小实例——用户输入区域标签 <input type="">.....33
2.5 HTML超链接标签.....20	2.10 实战应用——制作精美表格效果....34
2.5.1 属性href.....20	2.11 课后练习.....36
2.5.2 属性target.....21	
2.5.3 属性title.....22	
2.5.4 课堂小实例——链接到E-mail地址.....22	



第3章 数据类型和变量

3.1 基本数据类型	38
3.1.1 课堂小实例——使用字符串型数据	38
3.1.2 课堂小实例——使用数值型数据	39
3.1.3 课堂小实例——使用布尔型数据	39
3.1.4 课堂小实例——使用Undefined和Null类型	40
3.2 复合数据类型	40
3.2.1 常用的内置对象	40
3.2.2 课堂小实例——日期对象	40
3.2.3 数学对象	43
3.2.4 字符串对象	44
3.2.5 数组对象	45

3.3 常量	46
3.3.1 常量的种类	46
3.3.2 课堂小实例——常量的使用方法	46
3.4 变量	47
3.4.1 变量的含义	47
3.4.2 变量的定义方式	47
3.4.3 变量的命名规则	48
3.4.4 课堂小实例——变量的作用范围	48
3.5 实战应用——制作倒计时特效	49
3.6 课后练习	51

第4章 表达式与运算符

4.1 表达式	53
4.2 操作数	53
4.3 运算符介绍	54
4.3.1 运算符	54
4.3.2 操作数的类型	54
4.4 算术运算符	54
4.4.1 课堂小实例——加法运算符	54
4.4.2 课堂小实例——减法运算符	55
4.4.3 课堂小实例——乘法运算符	55
4.4.4 课堂小实例——除法运算符	56
4.4.5 课堂小实例——取模运算符	56
4.4.6 课堂小实例——负号运算符	56
4.4.7 课堂小实例——正号运算符	57
4.4.8 课堂小实例——递增运算符	57
4.4.9 课堂小实例——递减运算符	57
4.5 关系运算符	58
4.5.1 课堂小实例——相等运算符	58
4.5.2 课堂小实例——等同运算符	59
4.5.3 课堂小实例——不等运算符	59

4.5.4 课堂小实例——不等同运算符	60
4.5.5 课堂小实例——小于运算符	61
4.5.6 课堂小实例——大于运算符	61
4.5.7 课堂小实例——小于或等于运算符	61
4.5.8 课堂小实例——大于或等于运算符	62
4.6 课堂小实例——字符串运算符	62
4.7 赋值运算符	62
4.8 逻辑运算符	63
4.8.1 课堂小实例——逻辑与运算符	63
4.8.2 课堂小实例——逻辑或运算符	63
4.8.3 课堂小实例——逻辑非运算符	64
4.9 位运算符	64
4.9.1 课堂小实例——位与运算符	64
4.9.2 课堂小实例——位或运算符	65
4.9.3 课堂小实例——位异或运算符	65
4.9.4 课堂小实例——位非运算符	66
4.9.5 课堂小实例——左移运算符	66
4.9.6 课堂小实例——带符号右移运算符	67
4.10 课后练习	67

第5章 JavaScript程序核心语法

5.1 函数	69
5.1.1 什么是函数	69
5.1.2 理解函数的参数传递	70
5.1.3 函数中变量的作用域和返回值	70

5.2 函数的定义	71
5.2.1 函数的普通定义方式	71
5.2.2 函数的变量定义方式	72
5.2.3 函数的指针调用方式	73



5.3 使用选择语句	74
5.3.1 课堂小实例——if选择语句	74
5.3.2 课堂小实例——if...else选择语句	75
5.3.3 课堂小实例——if...else...if选择语句	76
5.3.4 课堂小实例——switch多条件选择语句	77
5.4 使用循环语句	78
5.4.1 课堂小实例——for循环语句	79

5.4.2 课堂小实例——while循环语句	79
5.4.3 课堂小实例——do-while循环语句	80
5.4.4 课堂小实例——break和continue 跳转语句	81
5.5 实战应用——禁止鼠标右击	83
5.6 课后练习	84

第6章 JavaScript核心对象

6.1 面向对象编程的简单概念	86
6.1.1 什么是面向对象	86
6.1.2 如何创建对象	87
6.1.3 对象的属性	87
6.1.4 对象的方法	87
6.2 对象应用	88
6.2.1 课堂小实例——声明和实例化	88
6.2.2 课堂小实例——对象的引用	90

6.2.3 对象的废除	91
6.2.4 课堂小实例——对象的早绑定和晚绑定	91
6.3 JavaScript的对象层次	91
6.3.1 客户端对象层次介绍	92
6.3.2 浏览器对象模型	92
6.4 综合实战——显示当前时间	93
6.5 课后练习	95

第7章 JavaScript中的事件

7.1 事件驱动与事件处理	97
7.1.1 事件与事件驱动	97
7.1.2 事件与处理代码关联	98
7.1.3 调用函数的事件	99
7.1.4 调用代码的事件	100
7.1.5 设置对象事件的方法	100
7.2 常见事件	101
7.2.1 课堂小实例——Click事件	101
7.2.2 课堂小实例——onchange事件	102

7.2.3 课堂小实例——onSelect事件	103
7.2.4 课堂小实例——onfocus事件	105
7.2.5 课堂小实例——onload事件	105
7.2.6 课堂小实例——鼠标移动事件	106
7.2.7 课堂小实例——onblur事件	108
7.3 其他常用事件	109
7.4 实战应用——将事件应用于按钮中	110
7.5 课后练习	111

第8章 window对象

8.1 window对象	113
8.1.1 window对象介绍	113
8.1.2 window对象的使用方法	114
8.2 窗口位置	116
8.2.1 课堂小实例——装载文档	116
8.2.2 课堂小实例——获取窗口外侧以及 内侧尺寸	116
8.2.3 课堂小实例——调整窗口的大小	118
8.3 对话框	118

8.3.1 课堂小实例——警告对话框	119
8.3.2 课堂小实例——询问对话框	119
8.3.3 课堂小实例——输入对话框	120
8.4 状态栏	121
8.4.1 状态栏介绍	121
8.4.2 课堂小实例——默认状态栏信息	121
8.4.3 课堂小实例——状态栏瞬间信息	121
8.5 窗口操作	122
8.5.1 课堂小实例——打开新窗口	122



8.5.2 课堂小实例——窗口名字	123
8.5.3 课堂小实例——关闭窗口	123
8.5.4 课堂小实例——窗口的引用	125

8.6 实战应用	125
8.7 课后练习	127

第9章 屏幕和浏览器对象

9.1 屏幕对象	130
9.1.1 课堂小实例——检测显示器参数	130
9.1.2 课堂小实例——客户端显示器屏幕 分辨率	131
9.1.3 课堂小实例——客户端显示器屏幕的 有效宽度和高度	132
9.1.4 课堂小实例——获取显示器的 显示信息	133

9.2 浏览器对象	133
9.2.1 课堂小实例——获取浏览器对象的 属性值	133
9.2.2 Plugin插件	134
9.3 综合实战	135
9.4 课后练习	137

第10章 文档对象

10.1 文档对象概述	140
10.1.1 文档对象介绍	140
10.1.2 文档对象的使用方法	143
10.1.3 课堂小实例——引用文档中对象的 方法	143
10.2 文档对象的应用	145
10.2.1 课堂小实例——设置超链接的颜色	145
10.2.2 课堂小实例——设置网页背景颜色和 默认文字颜色	145
10.2.3 课堂小实例——文档信息	146
10.2.4 课堂小实例——在网页中输出内容	147
10.2.5 课堂小实例——文档定位	148
10.2.6 课堂小实例——文档标题	149
10.2.7 课堂小实例——打开和关闭文档	149

10.3 链接对象	150
10.3.1 链接对象的介绍	150
10.3.2 课堂小实例——感知鼠标移动事件	151
10.4 脚本化cookie	151
10.4.1 cookie介绍	151
10.4.2 cookie的优点和缺点	152
10.4.3 检测浏览器是否支持cookie功能	152
10.4.4 创建cookie	153
10.4.5 cookie的生存期	153
10.5 实战应用——文字连续变换 多种颜色	153
10.6 课后练习	155

第11章 历史对象和地址对象

11.1 历史对象	157
11.1.1 历史对象的介绍	157
11.1.2 课堂小实例——前进到上一页和 后退到下一页	157
11.1.3 课堂小实例——跳转	158
11.1.4 课堂小实例——创建返回或前进到 数页前页面的按钮	159
11.2 地址对象	160
11.2.1 URL介绍	160

11.2.2 课堂小实例——获取当前页面 的URL	160
11.2.3 课堂小实例——加载新网页	162
11.2.4 课堂小实例——获取参数	162
11.2.5 重新装载当前文档	164
11.2.6 课堂小实例——加载新文档	164
11.2.7 课堂小实例——页面加载结束后， 加载下一个页面	166
11.3 实战应用	167
11.4 课后练习	168



第12章 表单对象和图片对象

12.1 Form表单对象	170
12.1.1 课堂小实例——在链接中使用 单选按钮	170
12.1.2 课堂小实例——给按钮添加链接	172
12.1.3 课堂小实例——给下拉菜单添加 链接	175
12.1.4 课堂小实例——在文本框中滚动 显示文字	177
12.1.5 课堂小实例——变更复选框的值	178
12.1.6 课堂小实例——密码验证	179
12.1.7 课堂小实例——确认是否重置	180
12.1.8 课堂小实例——选择上传的文件	182
12.2 image图片对象	183
12.2.1 课堂小实例——获取图片信息	183

12.2.2 课堂小实例——图片轮番显示效果	185
12.2.3 课堂小实例——控制动画播放	186
12.2.4 课堂小实例——指向或单击图片时， 使图片发生变换	188
12.2.5 课堂小实例——显示加载图片状态	190
12.2.6 课堂小实例——确认是否重新加载 图片	191
12.3 实战应用	192
12.3.1 实战应用1——如何制作在网页上 不断飘来飘去的图片	192
12.3.2 实战应用2——不用数据库，只有访问者输 入正确的名称与密码才能进入网页	194
12.4 课后练习	196

第13章 Ajax基础

13.1 了解传统的Web技术及Ajax的特点	198
13.2 认识Ajax技术的组成部分	199
13.2.1 Ajax中的JavaScript	199
13.2.2 Ajax中的XMLHttpRequest	199
13.2.3 Ajax中的CSS	200
13.2.4 Ajax中的DOM	200
13.2.5 Ajax中的XML	200

13.3 Ajax技术的优缺点分析	201
13.4 创建XMLHttpRequest	202
13.5 Ajax中Get请求与Post请求的区别	203
13.6 局部更新	205
13.7 一个完整的Ajax实例	208

第14章 导航菜单特效案例

14.1 横向展开的二级导航菜单	211
14.2 超实用的JavaScript下拉菜单	214
14.3 下拉折叠菜单	218

14.4 漂亮的分类导航菜单	224
14.5 网页滑动门菜单	225

第15章 文字和图片特效案例

15.1 JavaScript实现3D文字	230
15.2 飞翔的3D文字效果	231
15.3 3D旋转的文字	234
15.4 围绕鼠标转动的跟随文字	235

15.5 鼠标移上时放大图片	238
15.6 图片切换效果	239
15.7 网站横幅焦点图切换	243
15.8 百叶窗图片切换	246



第16章 按钮链接和页面特效案例

- | | | | |
|------------------------|-----|---------------------|-----|
| 16.1 背景图片变色的按钮..... | 251 | 16.6 网页定时刷新的特效..... | 258 |
| 16.2 单击按钮后按钮自动消失..... | 253 | 16.7 指定弹出窗口的位置..... | 259 |
| 16.3 闪烁的链接..... | 254 | 16.8 网页密码保护..... | 260 |
| 16.4 打开链接时弹出询问确认框..... | 256 | 16.9 鼠标滑过的列表..... | 261 |
| 16.5 在页面顶部显示进度条效果..... | 256 | | |

第17章 广告代码特效案例

- | | | | |
|-----------------------|-----|-------------------------|-----|
| 17.1 漂浮广告..... | 267 | 17.4 悬浮的QQ在线客服..... | 273 |
| 17.2 可关闭的对联广告..... | 269 | 17.5 多个图片广告交替显示..... | 278 |
| 17.3 收缩的Banner广告..... | 271 | 17.6 先显示大图随后自动收起显示小图... | 281 |

第1章

JavaScript 基础知识

本章导读

JavaScript是一种基于对象和事件驱动的客户端脚本语言。JavaScript最初的设计是为了检验HTML表单输入的正确性。JavaScript起源于Netscape公司的LiveScript语言。JavaScript几乎可以控制所有常用的浏览器，而且JavaScript是世界上最重要的编程语言之一，学习Web技术前必须要学会JavaScript。

技术要点

- ◎ JavaScript简介
- ◎ 在HTML中嵌入JavaScript的方法
- ◎ 第一个JavaScript程序



实例展示



在状态栏显示信息

1.1

JavaScript简介

JavaScript是一种脚本语言，比HTML要复杂。不过即便不懂编程，也不用担心，因为JavaScript写的程序都是以源代码的形式出现的，也就是说如果在一个网页里看到一段比较好的JavaScript代码，恰好也用得上，就可以直接拷贝，然后将其放到网页中去即可。

1.1.1 JavaScript的历史

JavaScript是Netscape公司与Sun公司合作开发的。在JavaScript出现之前，Web浏览器只能显示超文本文档软件的基本部分。而在JavaScript出现之后，网页的内容不再局限于枯燥的文本，它们的可交互性得到了显著的改善。JavaScript的第一个版本，即JavaScript 1.0版本，出现在1995年推出的Netscape Navigator 2浏览器中。

在JavaScript 1.0发布时，Netscape Navigator主宰着浏览器市场，微软的IE浏览器则扮演着追赶者的角色。微软在推出IE3的时候发布了自己的VBScript语言并以JScript为名发布了JavaScript的一个版本，以此很快跟上了Netscape的步伐。

面对微软公司的竞争，Netscape和Sun公司联合ECMA（欧洲计算机制造商协会）对JavaScript语言进行了标准化，其结果就是ECMAScript语言，这使得同一种语言又多了一个名字。虽说ECMAScript这个名字没有流行开来，但人们现在谈论的JavaScript实际上就是

ECMAScript。

到了1996年，JavaScript、ECMAScript、JScript——随便你们怎么称呼它，已经站稳了脚跟。Netscape和微软公司在它们各自的第3版浏览器中都不同程度地提供了对JavaScript 1.1语言的支持。

这里必须指出的是，JavaScript与Sun公司开发的Java程序语言没有任何联系。人们最初给JavaScript起的名字是LiveScript，后来选择“JavaScript”作为其正式名称的原因，大概是想让它听起来有系出名门的感觉，但令人遗憾的是，这一选择反而更容易让人们把这两种语言混为一谈，而这种混淆又因为各种Web浏览器确实具备这样或那样的Java客户端支持功能的事实被进一步放大和加剧。事实上，虽说Java在理论上几乎可以部署在任何环境中，但JavaScript却只局限于Web浏览器。

1.1.2 JavaScript特点

JavaScript具有以下语言特点。



- JavaScript是一种脚本编写语言,采用小程序段的方式实现编程,也是一种解释性语言,提供了一个简易的开发过程。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,同时也可以看作是一种面向对象的语言。这意味着它能运用自己已经创建的对象,因此它的许多功能可以来自于脚本环境中对象的方法与脚本的相互作用。
- JavaScript设计简单,首先它是一种基于Java基本语句和控制流之上的简单而紧凑的设计,其次它的变量类型采用弱类型,并未使用严格的数据类型。
- JavaScript是一种安全性语言,它不允许访问本地硬盘,并且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的丢失。

- JavaScript是动态的,它可以直接对用户或客户的输入做出响应,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,就是指在网页中执行了某种操作所产生的动作,就称为“事件”,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具有跨平台性。JavaScript是依赖于浏览器本身的,与操作环境无关,只要有能运行浏览器的计算机,并有支持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。

1.1.3 JavaScript和Java的区别

在各种网页制作技术论坛中,常常有人询问JavaScript与Java有什么区别,甚至有人误认为JavaScript就是Java。JavaScript与Java确实有一定的联系,但可以肯定地说它们之间有很大的区别。

1. 基于对象和面向对象

Java是SUN公司推出的新一代面向对象的程序设计语言,特别适合于Internet应用程序开发;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。

2. 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的束缚。

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。

3. 安全性

JavaScript作为网络的脚本语言,可以直接查看其源代码,所以安全等级不高,可以任

意复制粘贴。而Java应用在网页的程式称为Java Applet,是与HTML代码分开的,所以安全性相对较高。

4. 强变量和弱变量

JavaScript与Java相比,结构没有Java的严谨。

Java采用强类型变量检查,即所有变量在编译之前必须声明,如:

```
Integer x;  
String y;  
x=1234;  
y=4321;
```

其中x=1234说明是一个整数,y=4321说明是一个字符串。

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

```
x=1234;  
y="4321";
```

前者说明x为数值型变量,而后者说明y为字符型变量。

5. 代码格式不一样

Java是一种与HTML无关的格式,必须通过像HTML中引用外媒体那样进行装载,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。

6. 对文本和图形进行操作

JavaScript不直接对文本和图形进行操作，它在Web页面中与HTML元素组合在一起

发挥作用，但它可以控制浏览器，让浏览器直接对文本和图形进行处理。而Java则可以直接对文本和图形进行操作。

可以发现，JavaScript与Java虽然都可以应用于网页设计，但它们的确有太大的区别，根本就是两种语言。

1.2

在HTML中嵌入JavaScript的方法

JavaScript程序本身不能独立存在，它依附于某个HTML页面，在浏览器端运行。JavaScript本身作为一种脚本语言可以放在HTML页面中的任何位置，但是浏览器解释HTML时是按先后顺序的，所以放在前面的程序会被优先执行。

1.2.1 <script/>使用方法

在HTML中输入JavaScript代码时，需要使用<script>标签。在<script>标签中，language特性声明要使用的脚本语言，language特性一般被设置为JavaScript，不过也可用它声明JavaScript的确切版本，如JavaScript 1.3。

当浏览器载入网页Body部分的时候，就执行其中的JavaScript语句，执行之后输出的内容就显示在网页中。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>JavaScript语句</title>
</head>
<body>
<script type="text/javascript1.3">
<!--
var gt = unescape('%3e');
var popup = null;
var over = "Launch Pop-up Navigator";
popup = window.open("", 'popupnav', 'width=225,height=235,resizable=1,scrollbars=auto');
if (popup != null) {
if (popup.opener == null) {
popup.opener = self;
}
}
popup.location.href = 'tan.htm';
}
-->
</script>
</body>
</html>
```



浏览器通常忽略未知标签，因此在使用不支持JavaScript的浏览器阅读网页时，JavaScript代码也会被阅读。<!-- -->里的内容对于不支持JavaScript的浏览器来说就等同于一段注释，而对于支持JavaScript的浏览器，这段代码仍然会执行。

提示

通常JavaScript文件可以使用script标签加载到网页的任何一个地方，但是标准的方式是加载在head标签内。为防止网页加载缓慢，也可以把非关键的JavaScript放到网页底部。

通过指定script标签的src属性，就可以使用外部的JavaScript文件了。在运行时，这个js文件的代码全部嵌入到包含它的页面内，页面程序可以自由使用，这样就可以做到代码的复用。

提示

JavaScript文件外部调用的好处如下所述。

- ◆ 如果浏览器不支持JavaScript，将忽略script标签里面的内容，可以避免使用<!-- ... -->。
- ◆ 统一定义JavaScript代码，方便查看，方便维护。
- ◆ 使代码更安全，可以压缩、加密单个JavaScript文件。

1.2.2 外调脚本文件

如果很多网页都需要包含一段相同的代码，最好的方法是把这个JavaScript程序放到一个后缀名为.js的文本文件里。此后，任何一个需要该功能的网页，只需要引入这个js文件就可以了。

这样做，可以提高JavaScript的复用性，减少代码维护的负担，不必将相同的JavaScript代码拷贝到多个HTML网页里，将来一旦程序有所修改，也只要修改.js文件就可以。

在HTML文件中可以直接输入JavaScript，还可以将脚本文件保存在外部，通过<script>中的src属性指定URL来调用外部脚本语言。外部JavaScript语言的格式非常简单。事实上，它们只包含JavaScript代码的纯文本文件。在外部文件中不需要<script/>标签，引用文件的<script>标签出现在HTML页中，此时文件的后缀为“.js”。

```
<script type="text/javascript"
src="URL"></script>
```

实例代码：

```
<!doctype html>
<html>
<head>
<script src="http://www.baidu.com/
common.js"></script>
</head>
<body>
</body>
</html>
```

示例里的common.js其实就是一个文本文件，其内容如下：

```
function clickme()
{
    alert("You clicked me!")
}
```

1.2.3 直接位于事件处理部分的代码中

一些简单的脚本可以直接放在事件处理部分的代码中。如下所示，直接将JavaScript代码加入到OnClick事件中。

```
<input type="button" name="FullScreen" value="全屏显示"
onClick="window.open(document.location, 'big', 'fullscreen=yes')">
```

这里，使用<input/>标签创建一个按钮，单击它时调用onclick()方法。onclick特性声明一个事件处理函数，即响应特定事件的代码。

第一个JavaScript程序

1.3.1 预备知识

<!doctype html>	<!--HTML文档开始-->
<html>	<!--文档头开始-->
<head>	<!--设置文字字体-->
<meta charset="utf-8">	<!--标题开始-->
<title>	<!--标题结束-->
</title>	<!--文档头结束-->
</head>	<!--文档体开始-->
<body>	<!--脚本程序-->
<script language="JavaScript">	
alert("欢迎进入我的网页");	// JavaScript程序语句
//	// 更多的JavaScript程序语句
</script>	<!--脚本结束-->
</body>	<!--文档体结束-->
</html>	<!--HTML文档结束-->

1.3.2 JavaScript编辑器的选择

The screenshot shows a web browser window with the following content:

```

<html>
<head>
  <script language="JavaScript">
    ...
    alert("第一个JavaScript例子！");
    alert("JavaScript世界！");
    alert("学习JavaScript知识！");
  </script>
</head>
</html>

```

The browser's address bar displays the URL: `http://127.0.0.1:8000/1.html`.

图 1-1 JavaScript 代码编辑器

1.3.3 编写Hello World程序