

程序员
完全自学宝典

JavaScript 完全自学宝典



强锋科技 陈衍卿 编著

特点

从基本语法入手，起点低，通俗易懂
知识点按“原理讲解+实例应用”模式讲解
流行技术：DOM、Ajax、JSON、XML和Prototype等
代码注释详细，方便阅读

内容

基础：JavaScript的发展、基础语法、面向对象特性
DOM：表单控件、document和window对象
高级应用：正则表达式、cookie和UserData
最新技术：Ajax、XMLHttp、Prototype框架

强不息

无止境



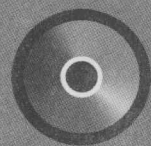
清华大学出版社

程序员
完全自学宝典

TP312/2920D

2008

完全自学宝典



JavaScript

强锋科技 陈衍卿 编著

自学
无止境

清华大学出版社
北 京

内 容 简 介

JavaScript 是一种基于对象与事件监听并具有安全性的动态脚本语言,也是目前使用最广泛的动态脚本语言之一。JavaScript 是最早出现的基于浏览器的动态脚本语言,也是目前几乎所有浏览器都支持的脚本语言。使用 JavaScript 可以创建具有高交互性与动态效果的 Web 网页,目前随着 Ajax 技术的发展以及在 Web 应用中的普及,JavaScript 成为目前最重要的程序设计语言之一。本书共分为 4 篇,从 JavaScript 的基础知识入手,由浅入深地介绍 JavaScript 的文档对象模型、JavaScript 中的各种高级应用以及 Ajax 与 JavaScript,循序渐进地介绍了 JavaScript 的相关知识及应用方法。

本书的特点是几乎所有知识点的介绍都使用了“原理讲解+实例应用”的方式,针对不同的知识点,分别介绍具有针对性并且实用性较强的实例,并针对某些较难理解的实例,做了不同程度的实例代码讲解。“原理+实例”的模式是最适合程序员学习与提高的学习方式。另外,本书部分实例按照从实际问题分析到设计直到最后代码实现的过程,使读者从解决问题的层面了解 JavaScript,从而达到“授人以渔”的根本目的。

本书适合具有一定 Web 基础(熟悉 HTML),没有 JavaScript 基础,但需要应用 JavaScript 或 Ajax 的程序员,也适合有一定的 JavaScript 但不了解 Ajax 的程序员,也非常适合没有 JavaScript 基础,但是对 JavaScript 非常感兴趣的初学者。

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 完全自学宝典/陈衍卿编著. —北京:清华大学出版社,2008.7
(程序员完全自学宝典)

ISBN 978-7-302-17641-1

I. J… II. 陈… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 073248 号

责任编辑:朱英彪 马子杰

封面设计:范华明

版式设计:魏 远

责任校对:王 云

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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:北京密云胶印厂

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185×260 印 张:34 字 数:762 千字

(附光盘 1 张)

版 次:2008 年 7 月第 1 版

印 次:2008 年 7 月第 1 次印刷

印 数:1~5000

定 价:59.80 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:027712-01

前 言

随着 Internet 的普及,各种 Web 应用飞速发展,目前 Web 应用软件已成为主流的软件系统。但是由于 Web 应用在交互性及动态展示中的缺陷,使得 Web 应用系统在客户体验方面,较传统 Client/Server 的应用系统略逊一筹。JavaScript 作为一种基于对象及事件监听的动态脚本语言,解决了部分交互性及动态 Web 页面的问题,随着 Ajax 技术的兴起及普及,JavaScript 成为一种重要的程序设计语言。目前在 JavaScript 与 Ajax 方面有很多书籍已出版,但是这些书籍要么只介绍 JavaScript,要么直接从 Ajax 入手更偏重于 Ajax,缺少一本从 JavaScript 基础开始到 JavaScript 应用然后过渡到 Ajax 应用的书籍。这些书籍不适合没有 JavaScript 基础的程序员快速应用 Ajax 技术。

本书从 JavaScript 的发展历史及几个具有特色的 JavaScript 实例开始,激发读者对 JavaScript 的兴趣,然后逐步深入地介绍 JavaScript 的基础知识、JavaScript 的文档结构树,以及 JavaScript 中的正则表达式、XML、浏览器插件等高级应用,最后介绍 JavaScript 与 Ajax 的相关知识及实例应用。读者在学习 JavaScript 知识的同时,可以了解几乎所有目前 JavaScript 实际应用中可能遇到的其他知识,学会对 JavaScript 的正确扩展,以便于解决实际中遇到的各种问题。

本书的内容安排

本书循序渐进地讲述了 JavaScript 相关知识,并由浅入深地介绍 JavaScript 的各种高级应用。本书共分为 4 篇。

第 1 篇(第 1~4 章)讲述了 JavaScript 作为一门程序设计语言的入门基础知识。包括 JavaScript 的发展历史、JavaScript 的特点、JavaScript 的语法基础、JavaScript 的高级语法以及 JavaScript 所具有的面向对象的特点。本部分在介绍基础知识的同时,列举了大量具有针对性的实例,供读者在了解理论的同时,通过实例加深对基础知识的理解。本部分是本书的基础,也是整个 JavaScript 的基础。

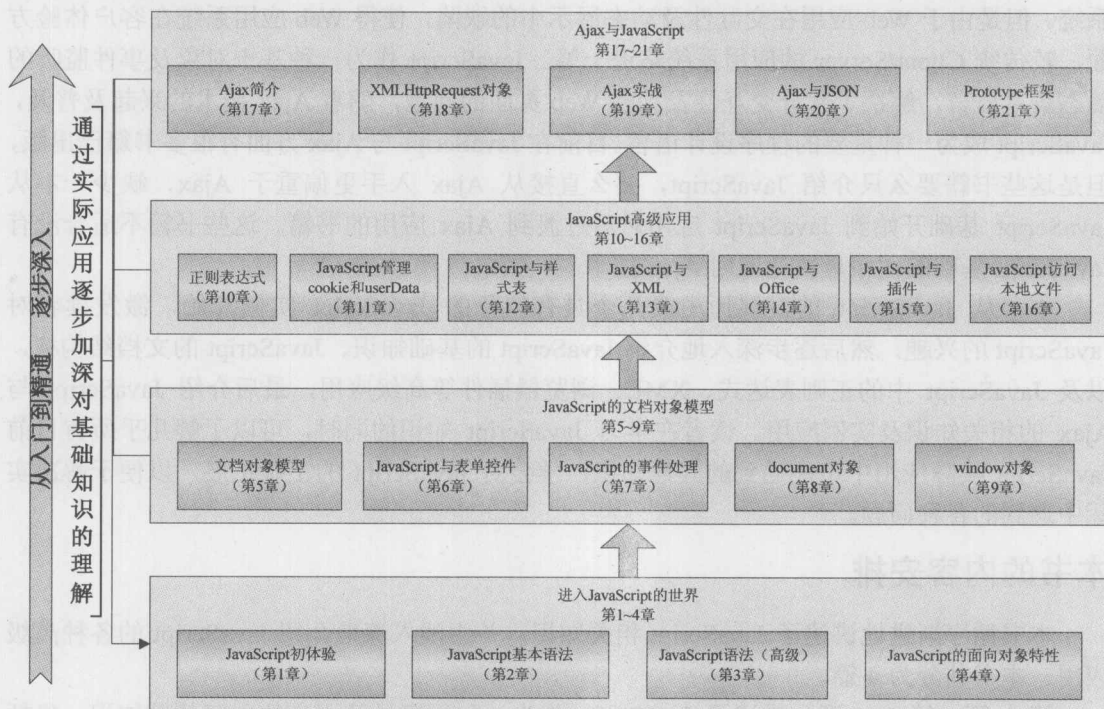
第 2 篇(第 5~9 章)讲述了 JavaScript 的文档对象模型。包括文档对象模型的概念、JavaScript 与表单控件、JavaScript 的事件处理、document 对象、window 对象。本部分是 JavaScript 高级应用的基础,并需要 JavaScript 的基础知识作为铺垫。

第 3 篇(第 10~16 章)介绍了 JavaScript 的高级应用。包括 JavaScript 中的正则表达式、JavaScript 管理 cookie 和 userData、JavaScript 与样式表、JavaScript 与 XML、JavaScript 与 Office、JavaScript 与插件、JavaScript 访问本地文件。本部分以前两部分做铺垫,包含了目前几乎所有的 JavaScript 实际应用(Ajax 除外)中可能遇到的问题,并结合大量实例及实例分析,介绍不同情况下使用 JavaScript 解决问题的思路。通过本部分,读者可以达到深入学习 JavaScript 并巩固 JavaScript 基础知识的目的。

第 4 篇(第 17~21 章)介绍了 Ajax 与 JavaScript。包括 Ajax 简介、XMLHttpRequest

对象、Ajax 实战、Ajax 与 JSON、Prototype 框架。本部分首先介绍目前普遍应用的 Ajax 技术的基础知识，其次通过各种 Ajax 应用实例，介绍具体应用中如何正确地在 Ajax 中使用 JavaScript。通过本部分，读者不仅可以了解并应用 Ajax 技术，还可以继续加深对 JavaScript 的理解，从而达到对 JavaScript 有较高层次的理解，并精通 JavaScript 的目的。

JavaScript 是一种相对容易掌握的程序设计语言，本书结合大量实例的讲解可以提高学习效率。读者可以遵循本书的写作顺序，参考以下流程阅读本书。



本书的特点

本书不仅包含了 JavaScript 语言本身的概念与基础知识，而且对实际应用 JavaScript 时可能涉及的正则表达式、XML、样式表、各种浏览器控件以及 Ajax 技术作了较为详细的讲解，同时针对不同的知识点介绍大量具有针对性并且实用的实例。另外，本书中的实例多数是笔者在实际开发中遇到的问题与经验的总结。本书将知识范围锁定在了适合初级、中级的部分，以大量的实例进行示范和解说。其特点主要体现在以下几个方面。

- ❑ 采用循序渐进的编排方式，适合初级、中级读者由浅入深地逐步掌握 JavaScript 语言。
- ❑ 重点介绍了 JavaScript 及其相关知识，为读者理解和实践奠定基础。
- ❑ 采用了大量的实例，结合实例加深对 JavaScript 各个知识点的理解。
- ❑ 所有实例都具有代表性和实际意义，着重解决工作中的实际问题。
- ❑ 针对较难理解的实例进行详细的解释和分析，帮助读者理解并形成分析问题的能力。

- ❑ 按递进关系进行案例组织，使用新旧知识点时相互关联，对比分析易于理解。
- ❑ 本书采用技术要点、详细介绍、运行效果等多种方式进行讲解，系统性强、可用性强。

适合阅读本书的读者

- ❑ 具有 HTML 基础的 JavaScript 及 Ajax 初学者。
- ❑ 具有一定 JavaScript 基础的 Ajax 技术初学者。
- ❑ 使用 JavaScript 进行 Web 应用开发的人员。
- ❑ 进行 Ajax 开发的人员。
- ❑ 各种门户网站开发人员。
- ❑ 本书尤其适合具有 HTML 基础的需要应用 JavaScript 与 Ajax 的程序员。

关于作者

本书由陈衍卿主持编写，其他参与编写、资料整理、代码调试的人员还有陈杰、陈冠军、项宇峰、于咏泽、冯浩楠、刘军、庞永庆、刘晶晶、刘辉、刘长江、吴荣、孙海民、孙爱荣、张亚丹、张军华、李家玉、李爱芝、李静、王全、王嘉、王晓天、王永刚、石光成、纪超、胡永、贾凯、赵美青、陆壮飞和马忠超等，在此一并表示感谢。

编 者

读者意见反馈卡

姓 名: _____ 性 别: ☐男 ☐女 职 业: _____
年 龄: _____ 文化程度: _____ E-mail: _____
通信地址: _____ 邮 编: _____
您常用的编程语言: 1 _____ 2 _____ 3 _____ 4 _____

您购买本书的原因 (可多选):

☐封面与装帧 ☐引言目录 ☐正文内容 ☐丛书风格 ☐价格 ☐光盘
☐专业性强 ☐别人介绍 ☐售后服务 ☐出版社或作者名声

本书最令您满意的是 (可多选):

☐专业性强、覆盖面广 ☐内容翔实、定位准确 ☐精益求精、售后服务

您可以承受的图书价格:

☐20 元以下 ☐30 元以下 ☐40 元以下 ☐50 元以下 ☐只要内容好, 不论价格

您对本书的评价:

封面装帧:	☐ 很好	☐ 较好	☐ 一般	☐ 不满意	建议 _____
印刷质量:	☐ 很好	☐ 较好	☐ 一般	☐ 不满意	建议 _____
正文质量:	☐ 很好	☐ 较好	☐ 一般	☐ 不满意	建议 _____
写作风格:	☐ 很好	☐ 较好	☐ 一般	☐ 不满意	建议 _____
专业水平:	☐ 很好	☐ 较好	☐ 一般	☐ 不满意	建议 _____

您希望增加哪些图书选题: 1 _____ 2 _____ 3 _____

您认为本书有哪些错误:

页码 _____	行 _____	列 _____	图号 _____	错误 _____	应改为 _____
页码 _____	行 _____	列 _____	图号 _____	错误 _____	应改为 _____
页码 _____	行 _____	列 _____	图号 _____	错误 _____	应改为 _____
页码 _____	行 _____	列 _____	图号 _____	错误 _____	应改为 _____

您的其他建议:

1 _____
2 _____
3 _____

请填写好本卡后寄给:

清华大学校内金地公司

邮编: 100084

电话: (010) 82899771

《JavaScript 完全自学宝典》编辑部收

传真: (010) 62788903

公司网址: www.thjd.com.cn

E-mail: zhuyingbiao@126.com

如需本书可与本编辑部联系邮购, 汇款请按以上地址填写, 另加邮费 15% (挂号)

目 录

第 1 篇 进入 JavaScript 的世界

第 1 章 JavaScript 初体验 2

1.1 JavaScript 简介 2

1.2 认识 JavaScript 2

1.2.1 JavaScript 的运行环境 3

1.2.2 JavaScript 代码的编写工具—— UltraEdit 3

1.2.3 JavaScript 制作最简单的 Hello World 实例 7

1.3 JavaScript 与 Java 的区别 8

1.4 JavaScript 与 HTML 9

1.4.1 在 HTML 网页头中嵌入 JavaScript 代码 9

1.4.2 在 HTML 网页体中嵌入 JavaScript 代码 10

1.4.3 在 HTML 网页的元素事件中 嵌入 JavaScript 代码 11

1.4.4 用 JavaScript 获取网页内容 实现数据验证 11

1.4.5 在 HTML 中调用已经存在的 JavaScript 文件 13

1.5 JavaScript 的错误调试 14

1.5.1 Internet Explorer 下的 Microsoft Script Editor 调试器 15

1.5.2 Firefox 下的 Venkman 调 试器 22

1.6 JavaScript 和浏览器 27

1.6.1 在 Internet Explorer 中调用 JavaScript 代码 27

1.6.2 在 Firefox 中调用 JavaScript 代码 28

1.6.3 在 Opera 中调用 JavaScript

代码 28

1.6.4 浏览器中的文档对象模型 (DOM) 30

1.7 JavaScript 的快乐体验 30

1.7.1 滚动字幕 30

1.7.2 数字时钟 34

1.7.3 浏览器的地震效果 35

1.8 小结 36

第 2 章 JavaScript 基本语法 37

2.1 基本语法结构 37

2.1.1 命名规则 37

2.1.2 注释 38

2.2 变量、关键字和保留字 38

2.2.1 变量 39

2.2.2 JavaScript 中的关键字 40

2.2.3 JavaScript 中的保留字 40

2.2.4 this 对象 40

2.2.5 with 语句 42

2.3 数据类型 43

2.3.1 string 字符串类型 43

2.3.2 number 数字类型 43

2.3.3 object 对象类型 44

2.3.4 Boolean 布尔类型 44

2.3.5 null 类型 44

2.3.6 undefined 类型 45

2.4 运算符 46

2.4.1 算术运算符 46

2.4.2 逻辑运算符 48

2.4.3 位运算符 49

2.4.4 赋值运算符 51

2.4.5 条件运算符 53

2.4.6 位操作运算符.....	54	3.2.5 数组头添加一项.....	87
2.4.7 字符串运算符.....	55	3.2.6 将数组对象内容反转.....	89
2.5 表达式.....	56	3.2.7 数组按字符的 ASCII 码 进行排序.....	89
2.5.1 赋值表达式.....	56	3.2.8 把数组各个项用某个字符串 连接起来.....	90
2.5.2 算术表达式.....	57	3.2.9 获取数组对象的所有内容.....	91
2.5.3 布尔表达式.....	57	3.2.10 多维数组.....	91
2.5.4 字符串表达式.....	59	3.3 数据类型转换.....	93
2.6 基本流程控制语句.....	60	3.3.1 转换为字符串.....	93
2.6.1 赋值语句.....	62	3.3.2 转换为数字.....	94
2.6.2 switch 选择语句.....	62	3.3.3 自动类型转换.....	95
2.6.3 while 循环语句.....	66	3.4 常用内置函数.....	96
2.6.4 for 循环语句.....	66	3.4.1 Math 数学对象.....	97
2.6.5 do while 循环语句.....	67	3.4.2 Date 日期对象.....	98
2.6.6 break 循环中止语句.....	67	3.4.3 编码函数 escape 和 unescape.....	100
2.6.7 continue 循环中断语句.....	68	3.4.4 动态执行代码 eval.....	100
2.7 函数.....	68	3.4.5 判断表达式是不是 数值 isNaN.....	101
2.7.1 函数的定义.....	69	3.5 小结.....	101
2.7.2 函数的调用.....	70		
2.7.3 将函数作为参数传递.....	70		
2.7.4 函数的隐含参数: arguments.....	71		
2.7.5 函数的 apply、call 方法和 length 属性.....	72		
2.8 小结.....	74		
第 3 章 JavaScript 语法 (高级).....	75		
3.1 字符串处理方法.....	75		
3.1.1 将两个或多个字符串组合 在一起.....	75		
3.1.2 找到子串在字符串中的位置.....	76		
3.1.3 查找指定位置的字符.....	76		
3.1.4 确认子串在字符串中最后 出现的位置.....	77		
3.1.5 截取字符串的一个子串.....	78		
3.1.6 提取字符串的一部分.....	79		
3.1.7 将一个字符串分割成一个 字符串数组.....	80		
3.2 数组.....	81		
3.2.1 计算数组的长度.....	81		
3.2.2 删除数组最后的项.....	82		
3.2.3 在数组末尾添加项.....	86		
3.2.4 删除数组第一项.....	86		
		第 4 章 JavaScript 的面向对象特性.....	102
		4.1 面向对象语言的特点.....	102
		4.2 JavaScript 中的面向对象开发.....	103
		4.2.1 对象的创建.....	103
		4.2.2 对象的调用.....	105
		4.2.3 对象的属性和方法.....	106
		4.2.4 对象的废除.....	106
		4.2.5 早绑定和晚绑定.....	107
		4.3 JavaScript 中对象的类型.....	107
		4.3.1 本地对象.....	107
		4.3.2 内置对象.....	108
		4.3.3 宿主对象.....	108
		4.3.4 作为关联数组的对象.....	108
		4.4 原型对象 prototype 和继承.....	111
		4.5 JavaScript 常用设计模式.....	114
		4.5.1 抽象工厂模式.....	114
		4.5.2 生成器模式.....	115
		4.5.3 适配器模式.....	116
		4.5.4 组合模式.....	117

4.6 异常处理	118	4.7 小结	124
----------------	-----	--------------	-----

第2篇 JavaScript 的文档对象模型

第5章 文档对象模型

126

5.1 什么是文档对象模型	126
5.2 理解 DOM 结构	127
5.3 使用 DOM 处理页面元素	133
5.3.1 获取页面的 body	134
5.3.2 获取元素的子节点	135
5.3.3 改变 div 的内容	137
5.3.4 改变页面的样式	139
5.4 使用 DOM 处理表格	140
5.4.1 获取表格的某行	140
5.4.2 获取表格的某列	143
5.4.3 修改表格的内容	146
5.4.4 设置表格的样式	148
5.4.5 无刷新效果更改表格 的某行	148
5.5 小结	153

第6章 JavaScript 与表单控件

154

6.1 什么是表单控件	154
6.2 文本框	156
6.2.1 获取文本框的值	157
6.2.2 JavaScript 控制文本框	159
6.2.3 文本框的 JavaScript 特效	162
6.3 列表框与组合框	168
6.3.1 获取列表框与组合框的值	169
6.3.2 JavaScript 控制列表框与 组合框	172
6.3.3 JavaScript 实现多级下拉 列表联动	177
6.4 复选框	182
6.4.1 获取复选框的值	182
6.4.2 JavaScript 控制复选框	184
6.4.3 复选框的 JavaScript 特效	188
6.5 单选按钮	189
6.5.1 获取单选按钮的值	190
6.5.2 JavaScript 控制单选按钮	191
6.5.3 单选按钮的 JavaScript 特效	193

6.6 按钮

195

6.6.1 JavaScript 控制按钮	195
6.6.2 重置按钮	197
6.6.3 提交按钮	197

6.7 小结

199

第7章 JavaScript 的事件处理

200

7.1 什么是事件处理

200

7.2 鼠标事件

201

7.2.1 onclick 单击事件	201
7.2.2 ondblclick 双击事件	202
7.2.3 onmouseover 鼠标移到 上方事件	204
7.2.4 onmouseout 鼠标离开事件	204
7.2.5 onmousedown 鼠标按下 事件	206

7.2.6 onmouseup 鼠标放开 按键事件	207
------------------------------------	-----

7.2.7 onselect 选中事件	208
---------------------------	-----

7.3 键盘事件

209

7.3.1 onkeydown 按键事件	209
7.3.2 onkeypress 按下键事件	209
7.3.3 onkeyup 放开键事件	210

7.4 HTML 事件

211

7.4.1 onload 窗口加载事件	211
7.4.2 onunload 窗口离开事件	212
7.4.3 onresize 改变窗口大小时 触发的事件	213

7.4.4 onabort 中断事件	215
--------------------------	-----

7.4.5 onerror 异常事件	216
--------------------------	-----

7.4.6 onreset 按下重置按钮事件	217
------------------------------	-----

7.4.7 onsubmit 按下提交按钮事件	218
-------------------------------	-----

7.5 变动事件

220

7.5.1 onblur 失去焦点事件	220
---------------------------	-----

7.5.2 onfocus 获得焦点事件	222
----------------------------	-----

7.5.3 onchange 值改变时 触发的事件	224
------------------------------------	-----

7.6 小结	224	第 9 章 window 对象	255
第 8 章 document 对象	225	9.1 window 对象属性	255
8.1 document 对象属性	225	9.1.1 默认的状态条消息	255
8.1.1 活动链接的颜色	225	9.1.2 当前显示的文档	257
8.1.2 文档锚点	226	9.1.3 窗口框架	258
8.1.3 文档锚点对象的数组	227	9.1.4 窗口框架的数组	260
8.1.4 修改文档的背景颜色	229	9.1.5 窗口的历史列表	260
8.1.5 存储于客户端的 cookie 信息	230	9.1.6 窗口内的框架数	261
8.1.6 设置文档的文本颜色	233	9.1.7 窗口所显示文档的完整 URL... ..	262
8.1.7 获取文档中的一个窗体	234	9.1.8 打开当前窗口的窗口	263
8.1.8 文档链接的颜色	235	9.1.9 当前页的父框架	264
8.1.9 文档标记的数组	236	9.1.10 当前框架的最顶层 浏览器窗口	266
8.1.10 当前显示文档的 URL	237	9.2 window 对象方法	267
8.1.11 获取包含链接的文档的 URL	239	9.2.1 打开消息框	267
8.1.12 用户已观察过的文档链接的 文本颜色	241	9.2.2 打开确认对话框	270
8.2 document 对象方法	242	9.2.3 打开输入对话框	272
8.2.1 清除指定文档的内容	242	9.2.4 打开一个新窗口	275
8.2.2 关闭文档流	245	9.2.5 把焦点从指定窗口移开	278
8.2.3 打开文档流	247	9.2.6 把指定的窗口带到前台	279
8.2.4 把文本写入文档	249	9.2.7 关闭窗口	280
8.2.5 把文本写入文档并换行	252	9.2.8 延时程序 (定时器)	281
8.3 小结	254	9.2.9 终止延时程序 (定时器) 的方法	283
		9.3 小结	284

第 3 篇 JavaScript 高级应用

第 10 章 正则表达式	286	10.2.1 句点符号	292
10.1 正则表达式语法	286	10.2.2 方括号符号	294
10.1.1 什么是正则表达式	286	10.2.3 “或”符号	295
10.1.2 正则表达式基本语法	286	10.2.4 表示匹配次数的符号	297
10.1.3 元字符	287	10.2.5 “否”符号	300
10.1.4 量词	287	10.2.6 圆括号和空白符号	303
10.1.5 字符类	289	10.2.7 其他符号	303
10.1.6 分组	290	10.3 RegExp 正则对象	304
10.1.7 反向引用	291	10.3.1 在字符串中查找满足表达式 的字符	304
10.1.8 候选	291	10.3.2 查找内容匹配的 第一个子字符串的位置	306
10.2 正则表达式符号	292		

10.3.3 判断某个字符串是否满足 表达式.....	307	12.3.1 外部定义样式的优先级	353
10.3.4 查找匹配正则表达式的 字符串.....	307	12.3.2 其他定义样式的优先级	357
10.4 常用正则表达式	307	12.4 CSS 属性	357
10.4.1 匹配数字格式.....	307	12.4.1 字体属性	357
10.4.2 匹配日期格式.....	309	12.4.2 颜色、背景属性	358
10.4.3 匹配时间格式.....	312	12.4.3 文本属性	358
10.4.4 匹配 HTML 标记	315	12.4.4 边框属性	359
10.4.5 匹配邮政编码.....	317	12.5 CSS 单位.....	360
10.4.6 匹配 IP 地址	319	12.5.1 长度单位	360
10.4.7 匹配 Email 地址	321	12.5.2 百分比单位	361
10.4.8 匹配网址 URL.....	324	12.5.3 颜色单位	363
10.4.9 匹配手机号码.....	326	12.6 CSS 使用	363
10.4.10 匹配中文字符.....	328	12.6.1 在当前页面中嵌入 CSS 样式代码	363
10.5 小结	330	12.6.2 在当前页面中嵌入 CSS 样式文件	364
第 11 章 JavaScript 管理 cookie 和 userData.....	331	12.6.3 动态修改页面元素的 CSS 样式	366
11.1 什么是 cookie 和 userData.....	331	12.7 实战练习: JavaScript 使用 CSS 模拟图片的圆角效果.....	370
11.2 JavaScript 操作 cookie	332	12.8 小结	372
11.2.1 保存数据到 cookie	332	第 13 章 JavaScript 与 XML.....	373
11.2.2 获取保存在 cookie 中的 数据.....	334	13.1 什么是 XML	373
11.2.3 修改 cookie 的保存有效期	336	13.2 使用 XPath.....	376
11.2.4 删除 cookie	338	13.2.1 数据模型	376
11.3 JavaScript 操作 userData.....	339	13.2.2 定位路径与定位步	378
11.3.1 保存数据到 userData.....	339	13.2.3 基本表达式	379
11.3.2 获取 userData 中的数据.....	342	13.2.4 函数调用	380
11.4 实战练习: 利用 cookie 登录实例	344	13.2.5 Internet Explorer 中的 XPath	382
11.5 小结	348	13.2.6 Firefox 中的 XPath.....	382
第 12 章 JavaScript 与样式表	349	13.2.7 跨浏览器兼容的 XPath	383
12.1 什么是样式表 (CSS) 与 为什么用 CSS.....	349	13.3 使用 XSLT	385
12.2 CSS 语法基础	350	13.3.1 简单实例: “Hello World” ...	385
12.2.1 注释	350	13.3.2 XSLT 的元素语法.....	387
12.2.2 选择符.....	351	13.4 XML 数据岛	388
12.3 CSS 优先级	353	13.4.1 什么是数据岛	388
		13.4.2 在 HTML 中显示 XML 数据	389

13.4.3 数据岛对象.....	391	15.3.1 什么是 Java Applet.....	416
13.4.4 JavaScript 操作数据岛	391	15.3.2 JavaScript 与 Java Applet 互操作	416
13.5 实战练习: JavaScript+XML 实现通讯录	394	15.4 Flash	424
13.6 小结	400	15.4.1 什么是 Flash.....	424
第 14 章 JavaScript 与 Office	401	15.4.2 JavaScript 与 Flash 互操作	424
14.1 JavaScript 操作 Office 的原理	401	15.5 JavaScript 操作 SVG	425
14.2 JavaScript 操作 Word.....	401	15.5.1 什么是 SVG	425
14.2.1 向 Word 中写入内容	402	15.5.2 JavaScript 与 SVG 互操作	426
14.2.2 从 Word 中读取内容	405	15.6 小结	428
14.3 JavaScript 操作 Excel.....	407	第 16 章 JavaScript 访问本地文件 ...	429
14.3.1 向 Excel 中写入内容	408	16.1 FileSystemObject 文件 操作对象.....	429
14.3.2 从 Excel 中读取内容	411	16.1.1 访问驱动器	430
14.4 小结	413	16.1.2 访问文件夹	432
第 15 章 JavaScript 与插件	414	16.1.3 访问文件	437
15.1 什么是插件	414	16.2 操作本地文件引起的 安全问题.....	440
15.2 ActiveX 控件.....	414	16.3 实战练习: 使用 FileSystemObject 操作本地文件实例.....	441
15.2.1 什么是 ActiveX 控件	414	16.4 小结	445
15.2.2 JavaScript 调用 ActiveX 控件	415		
15.3 Java Applet	416		

第 4 篇 Ajax 与 JavaScript

第 17 章 Ajax 简介	448	18.2.2 responseText 属性	459
17.1 Web 应用的发展历程	448	18.2.3 responseXML 属性	460
17.2 Ajax 的技术特点.....	451	18.2.4 status 属性	460
17.3 Ajax 的现状以及未来.....	452	18.2.5 statusText 属性	461
17.4 Ajax 与 Java	454	18.2.6 onreadystatechange 事件.....	461
17.5 Ajax 与 .Net.....	454	18.3 XMLHttpRequest 的方法	461
17.6 Ajax 的竞争者.....	455	18.3.1 abort()方法	461
17.7 小结	457	18.3.2 open()方法	462
第 18 章 XMLHttpRequest 对象	458	18.3.3 send()方法	462
18.1 什么是 XMLHttpRequest 对象.....	458	18.3.4 setRequestHeader()方法	462
18.2 XMLHttpRequest 的属性 和事件	459	18.3.5 getResponseHeader()方法	463
18.2.1 ReadyState 属性.....	459	18.3.6 getAllResponseHeaders() 方法	465
		18.4 发送请求	466
		18.5 处理请求	468

18.6 小结	469	20.2.2 值的数组	498
第 19 章 Ajax 实战	470	20.3 在 JavaScript 中使用 JSON	501
19.1 第一个 Ajax 程序: 显示 XMLHttpRequest 对象状态	470	20.3.1 将 JSON 数据赋值给变量	502
19.2 实例: 完成验证	472	20.3.2 访问 JSON 数据	503
19.2.1 完成验证的客户 (页面) 端	473	20.3.3 修改 JSON 数据	504
19.2.2 完成验证的服务器端	474	20.3.4 转换回字符串	504
19.3 实例: 动态加载组合框	476	20.4 JSON 实例: 使用 JSON 与 服务器传递信息	505
19.3.1 动态加载组合框的客户端 (页面)	477	20.5 小结	514
19.3.2 动态加载组合框的 服务器端	479	第 21 章 Prototype 框架	515
19.4 实例: 创建自动刷新页面	482	21.1 Prototype 框架简介	515
19.4.1 自动刷新网页的客户端 (页面)	482	21.2 实例: 用 \$() 方法获取 元素对象	516
19.4.2 自动刷新网页的服务器端	483	21.3 实例: 用 \$\$() 方法获取指定 元素数组	518
19.5 实例: 显示进度条	486	21.4 实例: 用 \$F() 方法获取指定 元素	519
19.5.1 显示进度条的客户端 (页面)	486	21.5 实例: 用 \$A() 方法获取数组	520
19.5.2 显示进度条的服务器端	489	21.6 实例: 用 \$H() 方法获取 Hash 对象	521
19.6 小结	492	21.7 实例: 用 \$R() 方法获取 取值范围	522
第 20 章 Ajax 与 JSON	493	21.8 实例: 用 Try.these() 方法 创建 XML DOM 对象	523
20.1 什么是 JSON	493	21.9 Ajax 对象	523
20.2 JSON 基础	494	21.10 Form 对象	525
20.2.1 简单 JSON 示例	495	21.11 小结	528

第 1 篇



DESIGN

进入 JavaScript 的世界

第 1 章 JavaScript 初体验

第 2 章 JavaScript 基本语法

第 3 章 JavaScript 语法（高级）

第 4 章 JavaScript 的面向对象特性

第 1 章 JavaScript 初体验

随着 Internet 的普及, 电子商务、电子政务、网上银行等各种网络应用正迅速发展。但与此同时, 随着个人电脑性能的不断提高、网络用户规模的逐渐扩大, 也带来了一些诸如个人电脑性能浪费、服务器压力过大、用户等待时间过长等问题。JavaScript 作为一种运行于客户端浏览器的、轻量的、解释性的并且基于对象与事件监听的脚本语言, 在这种情况下, 逐步受到重视并在实际中得到普遍应用。

本章将介绍 JavaScript 的基本概念、基本特征、应用技术基础(与 HTML 文件的融合、错误调试、浏览器支持)以及简单的 JavaScript 应用的实例。

1.1 JavaScript 简介

说到 JavaScript 不能不提的就是 Netscape(网景)公司, 这家公司的 Netscape 浏览器曾经风靡一时。Netscape 公司在设计浏览器时, 为提高网页的互动性, 开发了 LiveScript 编程语言, 而 LiveScript 还是 JavaScript 的前身。JavaScript 是 Netscape 公司与 Sun 公司合作、改进并重新设计 LiveScript 之后的产物。

随着技术的进步, JavaScript 的版本也在不断发展。到目前为止, 最新的 JavaScript 的版本是 JavaScript 1.7。目前支持 JavaScript 1.7 的浏览器有 Firefox 2.0.x 等。

目前, 在 Web 应用中, 使用相当普遍的技术就是 Ajax(本书将在后面章节中介绍)。Ajax 的组成中, 非常重要的技术就是 JavaScript, 这从 Ajax 的详细名称——Asynchronous JavaScript and XML(异步 JavaScript 和 XML)中可见一斑。

1.2 认识 JavaScript

概括地说, JavaScript 就是一种基于对象和事件驱动, 并具有安全性能的脚本语言。对象和事件是 JavaScript 的两个核心, 本书将在以后的章节详细介绍。

JavaScript 是一种脚本语言。脚本语言的特点是比较简单、易学, 即使是程序设计新手也可以非常容易地使用 JavaScript 进行简单的编程。

JavaScript 可以被嵌入到 HTML 文件中, 不需要经过 Web 服务器就可以对用户操作作出响应, 使网页更好地与用户交互; 在利用客户端个人电脑性能资源的同时, 适当减小服务器端的压力, 并减少用户等待时间。

说明: JavaScript 在 Web 浏览器端执行时, 不对客户端文件进行操作, 加大了系统的安全性。

JavaScript 的运行需要 JavaScript 解释器。换言之, 一个能运行 JavaScript 的 Web 浏览器, 需要嵌入 JavaScript 解释器。这与 Java 的只要有 Java 虚拟机就能运行、跨操作系统平台的特性类似。因此, JavaScript 的使用相当广泛。目前几乎所有的主流浏览器 (Internet Explorer、Firefox、Opera、Maxthon……) 都支持 JavaScript。

1.2.1 JavaScript 的运行环境

1. 软件环境

前文中提到 JavaScript 的运行需要 JavaScript 解释器的支持, 而与具体的操作系统无关。目前支持 JavaScript 的 Web 浏览器有 Internet Explorer、Firefox、Opera 和 Maxthon 等。另外, 各种 HTML、JavaScript 的 IDE 工具也属于 JavaScript 运行的软件环境。

2. 硬件环境

支持浏览器的操作系统, 要求最低硬件配置。

- ☐ CUP 至少 233MHz。
- ☐ 基本内存 32MB。
- ☐ 显示器至少 256 色, 分辨率 640×480 像素及以上。
- ☐ 鼠标等其他外设。

1.2.2 JavaScript 代码的编写工具——UltraEdit

JavaScript 的代码编辑器有很多, 如 Windows 系统内置的“记事本”工具、Microsoft FrontPage、Microsoft Script Editor、PrimalScript、Edit Plus、UltraEdit 和 Aptana 等。本节将介绍目前较为流行的, 同时也是使用比较方便的代码编写工具——UltraEdit。

1. 认识 UltraEdit

UltraEdit 是一款功能强大的代码文本编写工具, 由 IDM Computer Solutions 开发。UltraEdit 的功能有: 同时编辑多种类型的多个文件、十六进制编辑模式、列编辑模式、行号和列号显示、选中行高亮显示、选中的字节数、代码折叠、拼写检查、宏功能、文件格式指示、文件大小指示、文件比较、多文件的查找和替换、查找字符串下拉列表、用户定制菜单/工具栏、输出窗口、命令调用、FTP 支持以及强大的可扩展性等。UltraEdit 运行界面如图 1.1 所示。

本书介绍的 UltraEdit 版本为 12.10a, 支持的语言类型有 HTML、JavaScript、Perl、PHP、CSS、C/C++、Java、C#和 Visual Basic 等。UltraEdit 会自动根据文件类型, 选择合适的代码显示效果 (彩色显示), 用户可以根据需要自己选择合适的显示效果, 还可以自定义显示类型, 如图 1.2 和图 1.3 所示。