

### 图书基本信息

书名：<<JavaScript DOM编程艺术（第2版）>>

13位ISBN编号：9787115249999

10位ISBN编号：7115249997

出版时间：2011-4

出版单位：人民邮电出版社

作者：[英] Jeremy Keith,[加] Jeffrey Sambells

页数：286

译者：杨涛,王建桥,杨晓云 等

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## 内容概要

JavaScript是Web开发中最重要的一门语言，它强大而优美。  
无论是桌面开发，还是移动应用。  
JavaScript都是必须掌握的技术。  
W3C的DOM标准是开发Web应用的基石。  
已经得到所有现代浏览器的支持，这使得跨平台Web开发成了一件轻松惬意的事。

本书是超级畅销书的升级版，由倡导Web标准的领军人物执笔，揭示了前端开发的真谛，是学习JavaScript和DOM开发的必读之作。

本书在简洁明快地讲述JavaScript和DOM的基本知识之后，通过几个实例演示了专业水准的网页开发技术，透彻阐述了平稳退化等一批至关重要的JavaScript编程原则和最佳实践，并全面探讨了HTML5以及jQuery等JavaScript库。  
读者将看到JavaScript、HTML5和CSS如何协作来创建易用的、与标准兼容的Web设计，掌握使用JavaScript和DOM通过客户端动态效果和用户控制的动画来加强Web页面的必备技术；同时，还将对如何利用库提高开发效率有全面深入的理解。

## 作者简介

Jeremy Keith 国际知名的Web设计师，Web标准项目组成员，DOM Scripting任务组负责人之一，任职于Web咨询公司Clearleft。  
除本书外，还著有HTML5 for Web Designers、Bulletproof Ajax。  
可通过其个人网站adactio.com与他联系。

Jeffrey Sambells  
资深平面设计师，Web开发人员，创办了软件公司We-Create并担任研发总监。  
除本书外，还著有《JavaScript高级程序设计》和  
Beginning Google Maps Applications with PHP and Ajax等多部畅销书。

书籍目录

第1章 JavaScript简史

- 1.1 JavaScript的起源
- 1.2 DOM
- 1.3 浏览器战争
  - 1.3.1 DHTML
  - 1.3.2 浏览器之间的冲突
- 1.4 制定标准
  - 1.4.1 浏览器以外的考虑
  - 1.4.2 浏览器战争的结局
  - 1.4.3 崭新的起点
- 1.5 小结

第2章 JavaScript语法

- 2.1 准备工作
- 2.2 语法
  - 2.2.1 语句
  - 2.2.2 注释
  - 2.2.3 变量
  - 2.2.4 数据类型
  - 2.2.5 数组
  - 2.2.6 对象
- 2.3 操作
- 2.4 条件语句
  - 2.4.1 比较操作符
  - 2.4.2 逻辑操作符
- 2.5 循环语句
  - 2.5.1 while循环
  - 2.5.2 for循环
- 2.6 函数
- 2.7 对象
  - 2.7.1 创建对象
  - 2.7.2 宿主对象
- 2.8 小结

第3章 DOM

- 3.1 文档：DOM中的“D”
- 3.2 对象：DOM中的“O”
- 3.3 模型：DOM中的“M”
- 3.4 节点
  - 3.4.1 元素节点
  - 3.4.2 文本节点
  - 3.4.3 属性节点
  - 3.4.4 CSS
  - 3.4.5 获取元素
  - 3.4.6 盘点知识点
- 3.5 获取和设置属性
  - 3.5.1 getAttribute

3.5.2 setAttribute

3.6 小结

第4章 案例研究：JavaScript图片库

4.1 标记

4.2 JavaScript

4.2.1 非DOM解决方案

4.2.2 最终的函数代码清单

4.3 应用这个JavaScript函数

4.4 对这个函数进行扩展

4.4.1 childNodes属性

4.4.2 nodeType属性

4.4.3 在标记里增加一段描述

4.4.4 用JavaScript改变这段描述

4.4.5 nodeValue属性

4.4.6 firstChild和lastChild属性

4.4.7 利用nodeValue属性刷新这段描述

4.5 小结

第5章 最佳实践

5.1 过去的错误

5.1.1 不要怪罪JavaScript

5.1.2 Flash的遭遇

5.1.3 质疑一切

5.2 平稳退化

5.2.1 “ javascript: ” 伪协议

5.2.2 内嵌的事件处理函数

5.2.3 谁关心这个

5.3 向CSS学习

5.3.1 结构与样式的分离

5.3.2 渐进增强

5.4 分离JavaScript

5.5 向后兼容

5.5.1 对象检测

5.5.2 浏览器嗅探技术

5.6 性能考虑

5.6.1 尽量少访问DOM和尽量减少标记

5.6.2 合并和放置脚本

5.6.3 压缩脚本

5.7 小结

第6章 案例研究：图片?改进版

6.1 快速回顾

6.2 它支持平稳退化吗

6.3 它的JavaScript与HTML标记是分离的吗

6.3.1 添加事件处理函数

6.3.2 共享onload事件

6.4 不要做太多的假设

6.5 优化

6.6 键盘访问

6.7 把JavaScript与CSS结合起来

6.8 DOM Core和HTML-DOM

6.9 小结

## 第7章 动态创建标记

7.1 一些传统方法

7.1.1 document.write

7.1.2 innerHTML属性

7.2 DOM方法

7.2.1 createElement方法

7.2.2 appendChild方法

7.2.3 createTextNode方法

7.2.4 一个更复杂的组合

7.3 重回图片库

7.3.1 在已有元素前插入一个新元素

7.3.2 在现有方法后插入一个新元素

7.3.3 图片库二次改进版

7.4 Ajax

7.4.1 XMLHttpRequest对象

7.4.2 渐进增强与Ajax

7.4.3 Hijax

7.5 小结

## 第8章 充实文档的内容

8.1 不应该做什么

8.2 把“不可见”变成“可见”

8.3 内容

8.3.1 选用HTML、XHTML还是HTML5

8.3.2 CSS

8.3.3 JavaScript

8.4 显示“缩略语列表”

8.4.1 编写displayAbbreviations函数

8.4.2 创建标记

8.4.3 一个浏览器“地雷”

8.5 显示“文献来源链接表”

8.6 显示“快捷键清单”

8.7 检索和添加信息

8.8 小结

## 第9章 CSS-DOM

9.1 三位一体的网页

9.1.1 结构层

9.1.2 表示层

9.1.3 行为层

9.1.4 分离

9.2 style属性

9.2.1 获取样式

9.2.2 设置样式

9.3 何时该用DOM脚本设置样式

9.3.1 根据元素在节点树里的位置来设置样式

9.3.2 根据某种条件反复设置某种样式

9.3.3 响应事件

9.4 className属性

9.5 小结

第10章 用JavaScript实现动画效果

10.1 动画基础知识

10.1.1 位置

10.1.2 时间

10.1.3 时间递增量

10.1.4 抽象

10.2 实用的动画

10.2.1 提出问题

10.2.2 解决问题

10.2.3 CSS

10.2.4 JavaScript

10.2.5 变量作用域问题

10.2.6 改进动画效果

10.2.7 添加安全检查

10.2.8 生成HTML标记

10.3 小结

第11章 HTML5

11.1 HTML5简介

11.2 来自朋友的忠告

11.3 几个示例

11.3.1 Canvas

11.3.2 音频和视频

11.3.3 表单

11.4 HTML5还有其他特性吗

11.5 小结

第12章 综合示例

12.1 项目简介

12.1.1 原始资料

12.1.2 站点结构

12.1.3 页面结构

12.2 设计

12.3 CSS

12.3.1 颜色

12.3.2 布局

12.3.3 版式

12.4 标记

12.5 JavaScript

12.5.1 页面突出显示

12.5.2 JavaScript幻灯片

12.5.3 内部导航

12.5.4 JavaScript图片库

12.5.5 增强表格

12.5.6 增强表单

12.5.7 压缩代码

12.6 小结

附录 JavaScript库



## 章节摘录

版权页：1.4.1浏览器以外的考虑DOM是一种API（应用编程接口）。

简单地说,API就是一组已经得到有关各方共同认可的基本约定。

在现实世界中,相当于API的例子包括（但不限于）摩尔斯码、国际时区、化学元素周期表。

以上这些都是不同学科领域中的标准,它们使得人们能够更方便地交流与合作。

如果没有一个统一的标准,事情往往会演变成为一场灾难。

别忘了,因混淆英制度量衡与公制度量衡至少导致过一次火星探测任务的失败。

在软件编程领域中,虽然存在着多种不同的语言,但很多任务却是相同或相似的。

这也正是人们需要API的原因。

一旦掌握了某个标准,就可以把它应用在许多不同的环境中。

虽然语法会因为使用的程序设计语言而有所变化,但这些约定却总是保持不变的。

因此,虽然本书的重点是教会你如何通过JavaScript使用DOM,当？

需要使用诸如PHP或Python之类的程序设计语言去解析XML文档的时候,你获得的DOM新知识将会有很大的帮助。

W3C对DOM的定义是：“一个与系统平台和编程语言无关的接口,程序和脚本可以通过这个接口动态地访问和修改文档的内容、结构和样式。

”W3C推出的标准化DOM,在独立性和适用范围等诸多方面,都远远超出了各自为战的浏览器制造商们推出的各种专有DOM。

1.4.2浏览器战争的结局我们知道,浏览器市场份额大战中微软公司战胜了Netscape,具有讽刺意味的是,专有的DOM和HTML标记对这个最终结果几乎没有产生影响。

IE浏览器注定能击败其他对手,不过是因为所有运行Windows操作系统的个人电脑都预装了它。

受浏览器战争影响最重的人群是那些网站设计人员。

跨浏览器开发曾经是他们的噩梦。

除了刚才提到的那些在JavaScript实现方面的差异之外,Netscape Navigator和IE这两种浏览器在对CSS的支持方面也有许多非常不同的地方。

而编写那些可以同时支持这两种浏览器的样式表和脚本的工作也成了一种黑色艺术。

浏览器制造商的自私姿态遭到人们的激烈反对,一个名为Web标准计划的小组应运而生。

WaSP小组采取的第一个行动就是,鼓励浏览器制造商们采用W3C制定和推荐的各项标准,也就是在浏览器制造商们的帮助下得以起草和完善的那些标准。

或许是因为来自？

WaSP小组的压力,又或许是因为企业的内部决策,下一代浏览器产品对Web标准的支持得到了极大的改善。

1.4.3崭新的起点早期浏览器大战至今,浏览器市场已经发生了巨大的变化,而且到了今天,这一切也几乎每天都有变化。

有的浏览器,比如Netscape Navigator,差不多已经从人们的视野中消失了,而新一代浏览器则陆续登台亮相。

苹果公司在2003年首次发布了它的Safari浏览器（基于WebKit）,它从一开始就坚定不移地遵循DOM标准。

今天,包括Firefox、Chrome、Opera和IE,以及一些基于WebKit的其他浏览器都对DOM有着良好的支持。很多最潮的智能手机浏览器都在使用WebKit渲染引擎,推动着手持浏览器开发不断向前,让手机上网的体验甚至好过了使用某些桌面浏览器。

## 媒体关注与评论

本书不愧为经典。

文笔清新,深入浅出,不知不觉让你掌握优秀的编程原则,明白为什么要遵守标准。

——Slashdot我要隆重推荐本书。

它前所未有地演示了DOM脚本编程的真正潜力。

无论你是JavaScript新手还是专家,本书都绝对值得你拥有。

——Garrett Dimon,Digital-Web.com杂志专栏作家

编辑推荐

《JavaScript DOM编程艺术(第2版)》：Amazon超级畅销书最新释放JavaScript和DOM编程的惊人潜力  
涵养HTML5及jQuery

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>