

零壹快学程序设计系列丛书

书网合一

小白实战手册

零壹
快学

零基础 JavaScript 从入门到精通

立体化教学模式 零基础快速入门 零壹快学 编著

真人教学，手把手教你学JavaScript

- 丰富示例，贴近场景——丰富的代码示例，编程场景化
- 视频教学，动手操作——扫码即可学习配套视频，通俗易懂
- 线上问答，及时解惑——在线论坛，解答学习中遇到的疑问
- 海量题库，随时学习——大量题目练习，加快理解

SPM 南方出版传媒 广东人民出版社



零基础JavaScript 从入门到精通

零壹快学 编著

SPM 南方出版传媒 广东人民出版社

· 广州 ·

图书在版编目 (CIP) 数据

零基础JavaScript从入门到精通 / 零壹快学编著. —广州: 广东人民出版社, 2019.8
ISBN 978-7-218-13760-5

I. ①零… II. ①零… III. ①JAVA语言—程序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字 (2019) 第159428号

Ling Jichu JavaScript Cong Rumeng Dao Jingtong
零 基 础 J a v a S c r i p t 从 入 门 到 精 通
零壹快学 编著

 版权所有 翻印必究

出 版 人：肖风华

统筹策划：李婧玮

责任编辑：严耀峰

封面设计：画画鸭工作室

内文设计：奔流文化

责任技编：周 杰 吴彦斌

出版发行：广东人民出版社

地 址：广州市海珠区新港西路204号2号楼（邮政编码：510300）

电 话：（020）85716809（总编室）

传 真：（020）85716872

网 址：<http://www.gdpph.com>

印 刷：广东信源彩色印务有限公司

开 本：787毫米×1092毫米 1/16

印 张：40 字 数：800千

版 次：2019年8月第1版 2019年8月第1次印刷

定 价：90.00元

如发现印装质量问题，影响阅读，请与出版社（020-32449105）联系调换。

售书热线：020-32449123

零壹快学

《零基础 JavaScript 从入门到精通》编委会

主 编：郑显军

副 主 编：宋宏宇 杨 森

编委会成员：李延杰 蔡宇轩 吕鉴倬 汪心桐

廖 旭 吕 严

前言

历经七十多年的发展，无论是对于国内数以十万计的学习者而言，还是在有着多年培训经验的编者看来，学习编程语言，仍存在不小的难度，甚至有不少学习者因编程语言的复杂多变、难度太大而选择了中途放弃。实际上，只要掌握了其变化规律，即使再晦涩难懂的计算机专业词汇也无法阻挡学习者的脚步。对于初学者来说，若有一本能看得懂，甚至可以用于自学的编程入门书是十分难得的。为初学者提供这样一本书，正是我们编写本套丛书的初衷。

零壹快学以“零基础，一起学”为主旨，针对零基础编程学习者的需求和学习特点，由专业团队量身打造了本套计算机编程入门教程。本套丛书的作者都从事编程教育和培训工作多年，拥有丰富的教学经验，对于学习者常遇到的问题十分熟悉，在编写过程中针对这些问题花费了大量的时间和精力来加以阐释，对书中的每个示例反复推敲，加以取舍，按照学习者的接受程度雕琢示例涉及的技术点，力求成就一套真正适合初学者的编程书籍。

本套丛书涵盖了Java、PHP、Python、JavaScript、HTML、CSS、Linux、iOS、C语言、C++、C#等计算机语言，同时借助大数据和云计算等技术，为广大编程学习者提供计算机各学科的视频课程、在线题库、测评系统、互动社区等学习资源。

◆ 课程全面，聚焦实战

本套丛书涵盖多门计算机语言，内容全面、示例丰富、图文并茂，通过通俗易懂的语言讲解相关计算机语言的特性，以点带面，突出开发技能的培养，既方便学习者了解基础知识点，也能帮助他们快速掌握开发技能，为编程开发设计积累实战经验。

◆ 专业团队，紧贴前沿

本套丛书作者由一线互联网公司高级工程师、知名高校教师和研究所技术人员等组成，线上线下同步进行专业讲解及点评分析，为学习者扫除学习障碍。与此同时，团队在内容研发方向上紧跟当前技术领域热点，及时更新，直击痛点和难点。

◆ 全网覆盖，应用面广

本套丛书已全网覆盖Web、APP和微信小程序等客户端，为广大学习者提供包括计算机编程、人工智能、大数据、云计算、区块链、计算机等级考试等在内的多门视频课程，配有相关测评系统和技术交流社区，互动即时性强，可实现在线教育随时随地轻松学。

JavaScript是最流行的Web编程语言之一。它是一种基于原型链和事件驱动的跨平台解释型执行语言，广泛应用于Web网站开发，嵌套在HTML页面中，能够直接在电脑、电视、手机、平板等终端设备上运行。

JavaScript语言易学易用，受到浏览器厂商和众多开发者的喜爱，从诞生到现在牢牢占据着Web前端编程语言的第一位。本书以编程初学者的视角，详细介绍了JavaScript从入门到精通需要掌握的技术知识，同时特别指出容易出现误解的知识点，希望能够帮助读者快速入门，并掌握JavaScript这门语言的特性。

• 本书内容

◆ **基础知识**：第1~4章，主要介绍了JavaScript的概况、基础语法和基本语句，帮助读者打好基础，快速进入JavaScript的学习之旅。

◆ **核心编程**：第5~12章，主要介绍JavaScript编程的主要知识点，涵盖运算、流程控制、函数、对象、数组、字符串、正则表达式、原型链等，帮助读者掌握JavaScript的重点知识和设计理念。

◆ **浏览器编程**：第13~17章，主要介绍JavaScript主要运行环境——浏览器的相关知识，涵盖BOM、DOM、事件处理、表单、AJAX等，帮助读者理解与JavaScript密切相关的浏览器环境，学会处理实际应用中可能遇到的疑难问题。

◆ **进阶应用**：第18~22章，将JavaScript与HTML、CSS完整结合，介绍了HTML5的进阶知识点和流行框架，帮助读者掌握使用JavaScript构建应用的技能，向更高层次的JavaScript应用开发迈进。

◆ **项目实战**：第23章，以完整的项目案例演示JavaScript的实战开发过程，增强读者的实战能力、分析和解决问题的能力。

• 本书特点

◆ **由浅入深，全面讲解**。本书以编程初学者的视角，循序渐进地讲解JavaScript的语法知识，让初学者逐步掌握JavaScript的语法规则和编程思想。

◆ **示例丰富，贴近场景**。本书提供了丰富的代码示例，而且每段代码后都有解

释，便于读者清晰理解代码的含义。这些示例大多选自工作中的各类场景，力求做到编程场景化，让读者可以感受到真实的企业编程，提高分析解决问题的能力，增加实战操作经验。

◆ **视频教学，动手操作。**本书每一章都配有教学视频，直观展示了JavaScript代码运行的效果，并配有通俗易懂的解释。

◆ **知识拓展，难度提升。**本书在每一章末尾设有“小结”和“知识拓展”。通过在“知识拓展”中列举一些重要或有一定难度的知识点，为感兴趣的读者提供更多的拓展类学习内容，多维度强化自身的学习，加深对JavaScript的理解。

◆ **线上问答，及时解惑。**本书为确保广大读者的学习能够顺利进行，提供了在线答疑服务，希望通过这种方式及时解决各位读者在学习JavaScript的过程中遇到的困难和疑惑。

• 本书配套资源（可扫下方二维码获取）

- ◆ **大量的代码示例。**通过运行这些代码，读者可以进一步巩固所学的知识。
- ◆ **零壹快学官方视频教程。**力求让读者学以致用，知行并进，加强实战能力。
- ◆ **在线答疑。**为读者解惑，帮助读者解决学习中的困难，快速掌握要点难点。

• 本书适用对象

- ◆ 编程的初学者、爱好者与自学者
- ◆ 高等院校和培训学校的师生
- ◆ 职场新人
- ◆ 准备进入互联网行业的再就业人群
- ◆ “菜鸟”程序员
- ◆ 初、中级程序开发人员



零壹快学微信公众号

《零基础JavaScript从入门到精通》从初学者角度出发，详细讲述了JavaScript编程语言的基础知识点和应用开发所需要掌握的技能。全书内容通俗易懂，代码示例丰富，步骤清晰，图文并茂，可以使读者轻松掌握JavaScript的精髓，是一本实用的JavaScript入门书。

编者

2019年7月

目 录

CONTENTS

第 1 章 走进 JavaScript	1	2.2 外联 JavaScript	20
1.1 JavaScript 简介	1	2.2.1 内联 JavaScript	20
1.1.1 JavaScript 的起源	1	2.2.2 外联 JavaScript	20
1.1.2 JavaScript 的特点	2	2.2.3 行内 JavaScript	23
1.1.3 ECMAScript 6	3	2.3 执行 JavaScript 程序	24
1.1.4 ECMAScript 版本发布	3	2.3.1 代码块	24
1.2 JavaScript 开发环境	4	2.3.2 变量提升	27
1.2.1 Sublime Text 下载和安装	4	2.3.3 输出结果	28
1.2.2 WebStorm 下载和安装	6	2.3.4 动态执行	29
1.3 JavaScript 运行环境	7	2.3.5 延迟执行	32
1.3.1 浏览器	7	2.4 小结	33
1.3.2 Node.js	7	2.5 知识拓展	33
1.4 学好 JavaScript 的建议	8	2.5.1 JavaScript 框架	33
1.5 小结	8	2.5.2 JavaScript 压缩	34
1.6 知识拓展	9	第 3 章 JavaScript 语法	36
1.6.1 常用软件资源	9	3.1 语法基础	36
1.6.2 浏览器兼容性	10	3.1.1 标识符	37
1.6.3 开发者社区	11	3.1.2 直接量	38
第 2 章 初次使用 JavaScript	12	3.1.3 变量	40
2.1 Hello, JavaScript	12	3.1.4 常量	41
2.1.1 编写脚本	12	3.1.5 注释	43
2.1.2 执行脚本	18	3.1.6 保留字	43
2.2 编写 JavaScript 程序	19	3.2 数据类型	44
2.2.1 内联 JavaScript	19	3.2.1 布尔值	44
		3.2.2 数字	45
		3.2.3 字符串	47

3.2.4	对象	48
3.2.5	空值 null	49
3.2.6	未定义 undefined	50
3.2.7	使用 typeof 获取数据类型	51
3.3	严格模式	53
3.3.1	开启严格模式	53
3.3.2	严格模式的限制	54
3.4	小结	55
3.5	知识拓展	55
3.5.1	未声明与未定义的区别	55
3.5.2	JavaScript 编程风格	57
3.5.3	JavaScript 文档工具	58
第 4 章	JavaScript 运算符	59
4.1	运算符	59
4.2	算术运算符	62
4.2.1	加减乘除	62
4.2.2	递增递减	63
4.2.3	取余取负	65
4.2.4	幂运算	66
4.3	比较运算符	68
4.3.1	相等与全等	68
4.3.2	类型转换	69
4.3.3	大小判断	72
4.4	逻辑运算符	74
4.5	位运算符	77
4.5.1	按位逻辑运算	78
4.5.2	位移运算	78
4.6	赋值运算符	79
4.7	对象操作运算符	80
4.7.1	new 运算符	80
4.7.2	delete 运算符	82
4.7.3	点号运算符	85
4.7.4	中括号运算符	87
4.8	其他运算符	89
4.8.1	条件运算符	89

4.8.2	逗号运算符	89
4.8.3	小括号运算符	90
4.8.4	in 运算符	90
4.8.5	instanceof 运算符	92
4.8.6	void 运算符	93
4.9	运算符优先级	93
4.10	小结	95
4.11	知识拓展	95
4.11.1	检测运算顺序	95
4.11.2	惰性运算	96
第 5 章	JavaScript 数值运算	98
5.1	特殊数值	98
5.1.1	最大值 / 最小值	98
5.1.2	无穷大 / 极小值	101
5.1.3	负零	104
5.1.4	非数字 NaN	104
5.2	进制	105
5.2.1	二进制	105
5.2.2	八进制	106
5.2.3	十进制	107
5.2.4	十六进制	108
5.3	内置函数	109
5.3.1	parseInt	109
5.3.2	parseFloat	112
5.4	小结	112
5.5	知识拓展	112
5.5.1	进制转换	112
5.5.2	浮点数误差	113
5.5.3	三十六进制	114
第 6 章	JavaScript 流程控制	115
6.1	基本语句	115
6.1.1	声明语句	115
6.1.2	表达式语句	116
6.1.3	复合语句	116

6.1.4 空语句	118	7.4.1 简单调用	164
6.2 条件控制	119	7.4.2 方法调用	165
6.2.1 if 语句	119	7.5 函数返回值	166
6.2.2 switch 语句	124	7.6 嵌套函数	168
6.2.3 if 和 switch 对比	127	7.7 变量作用域	169
6.2.4 条件嵌套	129	7.7.1 全局作用域	169
6.3 循环控制	131	7.7.2 函数作用域	171
6.3.1 for/for...in 语句	131	7.7.3 变量优先级	172
6.3.2 while/do...while 循环语句	134	7.8 闭包函数	174
6.3.3 for 和 while 对比	136	7.9 this 对象	176
6.3.4 循环语句嵌套	136	7.10 小结	180
6.4 跳转控制	137	7.11 知识拓展	180
6.4.1 标签语句	137	7.11.1 动态参数	180
6.4.2 break 语句	138	7.11.2 递归函数	181
6.4.3 continue 语句	140	第 8 章 JavaScript 对象	183
6.5 异常控制	142	8.1 对象介绍	183
6.5.1 try...catch...finally 语句	142	8.2 创建对象	187
6.5.2 throw 语句	145	8.2.1 使用花括号创建对象	187
6.6 小结	146	8.2.2 使用 Object() 创建对象	188
6.7 知识拓展	147	8.2.3 使用 Object.create() 创建对象	190
6.7.1 避免使用死循环	147	8.3 对象属性	191
6.7.2 避免使用标签语句	147	8.3.1 定义属性	191
第 7 章 JavaScript 函数	148	8.3.2 配置属性	194
7.1 函数定义	148	8.3.3 赋值属性	199
7.1.1 function 关键字	148	8.3.4 删除属性	200
7.1.2 使用 Function() 构造函数	152	8.4 对象操作	201
7.1.3 匿名函数	153	8.4.1 访问对象	201
7.2 函数属性	155	8.4.2 枚举对象	202
7.2.1 name 属性	155	8.4.3 克隆对象	203
7.2.2 length 属性	157	8.4.4 销毁对象	204
7.3 函数参数	158	8.5 内置对象	205
7.3.1 形参和实参	158	8.5.1 Date 日期对象	205
7.3.2 传值和传引用	159	8.5.2 Math 数学对象	208
7.3.3 arguments 对象	161	8.6 小结	210
7.4 函数调用	164	8.7 知识拓展	210

8.7.1 原始值	210	10.4.3 连接和拆分	248
8.7.2 冻结对象	211	10.4.4 字符串转义	249
第 9 章 JavaScript 数组	213	10.4.5 大小写转换	251
9.1 数组介绍	213	10.4.6 字符串转 HTML 函数	252
9.2 数组定义	215	10.5 小结	252
9.2.1 使用中括号定义数组	215	10.6 知识拓展	252
9.2.2 使用 Array() 构造函数	216	10.6.1 模板字符串	252
9.2.3 二维数组	217	10.6.2 Unicode 转义序列	255
9.3 数组属性	218	第 11 章 JavaScript 正则表达式	257
9.3.1 length 属性	218	11.1 正则表达式介绍	257
9.3.2 prototype	220	11.2 正则表达式定义	258
9.4 数组操作	221	11.2.1 使用双斜线定义	258
9.4.1 新增元素	222	11.2.2 使用 RegExp() 构造函数	258
9.4.2 获取元素	224	11.3 正则表达式语法	259
9.4.3 查找元素	224	11.3.1 基本字符	260
9.4.4 删除元素	227	11.3.2 字符转义	260
9.4.5 替换元素	228	11.3.3 元字符	261
9.4.6 数组迭代	229	11.3.4 限定符	263
9.4.7 数组排序	232	11.3.5 定位符	265
9.5 小结	233	11.3.6 修饰符	267
9.6 知识拓展	234	11.3.7 分组捕获	269
9.6.1 数组求和	234	11.3.8 优先级	270
9.6.2 快速交换	234	11.3.9 注释	271
第 10 章 JavaScript 字符串	235	11.4 正则表达式函数	271
10.1 字符串介绍	235	11.4.1 regexp.exec()	271
10.2 字符串定义	235	11.4.2 regexp.test()	272
10.2.1 字符串常量	236	11.4.3 string.match()	273
10.2.2 字符串对象	237	11.4.4 string.replace()	274
10.3 字符串属性	239	11.4.5 string.search()	275
10.3.1 length 属性	239	11.4.6 string.split()	276
10.3.2 prototype 属性	240	11.5 小结	277
10.4 字符串操作	241	11.6 知识拓展	277
10.4.1 查找和替换	241	11.6.1 预查匹配	277
10.4.2 字符串截取	244	11.6.2 贪婪匹配	279
		11.6.3 常用正则表达式	280

第 12 章 原型链	286	13.6.4 关闭窗口	328
12.1 原型链介绍	286	13.6.5 窗口位置	329
12.2 原型属性	287	13.6.6 缩放事件	331
12.2.1 prototype 属性	287	13.6.7 异步调用	332
12.2.2 __proto__ 属性	290	13.7 Notification	335
12.3 原型扩展	291	13.7.1 请求权限	335
12.3.1 扩展原型	291	13.7.2 显示通知	336
12.3.2 修改原型	292	13.8 小结	337
12.4 原型继承	294	13.9 知识拓展	337
12.4.1 私有方法	295	13.9.1 防止网页嵌套	337
12.4.2 公有方法	296	13.9.2 BOM 跨域限制	339
12.4.3 实例方法	296	第 14 章 DOM 文档对象模型	341
12.5 小结	297	14.1 DOM 介绍	341
12.6 知识拓展	298	14.2 DOM 节点	343
12.6.1 原型陷阱	298	14.2.1 节点分类	343
12.6.2 继承方式	298	14.2.2 节点对象	345
第 13 章 BOM 浏览器对象模型	306	14.2.3 节点关系	349
13.1 BOM 介绍	306	14.3 节点属性	350
13.2 document 文档对象	307	14.3.1 获取属性	350
13.2.1 对象集合	307	14.3.2 设置属性	352
13.2.2 对象属性	309	14.3.3 删除属性	354
13.2.3 对象方法	311	14.3.4 数据属性	355
13.3 location 导航对象	313	14.4 元素操作	356
13.3.1 网页网址	313	14.4.1 访问元素	356
13.3.2 网页访问	314	14.4.2 遍历元素	358
13.4 history 历史对象	315	14.4.3 新建元素	360
13.4.1 网页操作	315	14.4.4 插入元素	361
13.4.2 网页刷新	316	14.4.5 查找元素	365
13.5 navigator 浏览器对象	317	14.4.6 复制元素	367
13.5.1 浏览器信息	318	14.4.7 删除元素	368
13.5.2 多语言与本地化	320	14.4.8 替换元素	369
13.6 window 窗口对象	323	14.5 节点字符串	370
13.6.1 全局作用域	323	14.5.1 文本节点	371
13.6.2 对话框	324	14.5.2 插入文本节点	371
13.6.3 新建窗口	327	14.5.3 节点 HTML	373

14.5.4 插入 HTML	374	15.7.2 触摸移动	421
14.6 样式表	376	15.7.3 触摸结束	422
14.6.1 获取样式	376	15.7.4 触摸取消	423
14.6.2 设置样式	380	15.8 事件模拟	424
14.6.3 动画效果	381	15.8.1 创建事件	424
14.7 小结	384	15.8.2 触发事件	425
14.8 知识拓展	384	15.9 小结	426
14.8.1 浏览器重绘	384	15.10 知识拓展	426
14.8.2 浏览器重排	385	15.10.1 事件委托	426
第 15 章 事件处理	387	15.10.2 自定义事件	430
15.1 事件介绍	387	第 16 章 表单对象	432
15.1.1 什么是事件	387	16.1 表单元素	432
15.1.2 事件冒泡	390	16.1.1 表单对象	432
15.1.3 事件捕获	392	16.1.2 控件列表	436
15.2 事件属性	393	16.1.3 控件属性	439
15.2.1 基本属性	393	16.2 表单事件	441
15.2.2 鼠标事件属性	395	16.2.1 获取焦点	441
15.2.3 键盘事件属性	396	16.2.2 失去焦点	444
15.3 事件方法	398	16.2.3 内容修改	445
15.3.1 停止冒泡	398	16.2.4 提交事件	445
15.3.2 停止同级冒泡	400	16.3 表单提交	447
15.3.3 阻止默认操作	402	16.3.1 使用 GET 提交	447
15.4 页面事件	403	16.3.2 使用 POST 提交	447
15.4.1 网页加载	403	16.3.3 文件上传	448
15.4.2 资源加载	406	16.4 小结	448
15.4.3 网页滑动	407	16.5 知识拓展	449
15.4.4 网页卸载	408	16.5.1 表单序列化	449
15.4.5 标签事件	410	16.5.2 无刷新提交	451
15.5 键盘事件	412	第 17 章 AJAX 异步通信	453
15.6 鼠标事件	414	17.1 AJAX 介绍	453
15.6.1 鼠标点击	414	17.1.1 异步提交	453
15.6.2 鼠标移动	416	17.1.2 XML 语言	453
15.6.3 拖拽元素	418	17.1.3 XMLHttpRequest 对象	454
15.7 触摸事件	419	17.1.4 NGINX 服务器配置	456
15.7.1 触摸开始	420		

17.2 XML 数据	457	第 19 章 数据存储	493
17.2.1 XML 文档	457	19.1 Cookie	493
17.2.2 XML 解析	457	19.1.1 Cookie 获取	494
17.3 JSON 数据	458	19.1.2 Cookie 设置	495
17.3.1 JSON 文档	458	19.1.3 Cookie 删除	496
17.3.2 JSON 解析	459	19.1.4 Cookie 安全	497
17.3.3 JSON 与 XML 对比	462	19.2 Session	497
17.4 AJAX 应用	462	19.3 WebStorage	498
17.4.1 XMLHttpRequest 初始化	462	19.3.1 localStorage 对象	498
17.4.2 设置 readystatechange	464	19.3.2 sessionStorage 对象	502
17.4.3 设置 HTTP 请求头信息	465	19.3.3 Cookie 与 Storage 对比	503
17.4.4 发送请求	467	19.4 小结	504
17.4.5 获取 HTTP 响应头信息	468	第 20 章 多媒体	505
17.4.6 获取响应文本	469	20.1 文件	505
17.4.7 获取 JSON 数据	469	20.1.1 Blob 对象	505
17.4.8 获取 XML 数据	471	20.1.2 File 对象	506
17.4.9 获取二进制数据	472	20.1.3 读取文件	508
17.5 小结	473	20.2 画板	513
17.6 知识拓展	473	20.2.1 canvas 标签	513
17.6.1 请求进度条	473	20.2.2 canvas 绘图	515
17.6.2 跨域请求	476	20.3 音频	517
第 18 章 CSS 网页特效	478	20.3.1 audio 标签	517
18.1 CSS 操作	478	20.3.2 audio 属性	518
18.1.1 样式对象	478	20.4 视频	520
18.1.2 计算样式	480	20.4.1 video 标签	520
18.1.3 设置样式	481	20.4.2 video 属性	520
18.2 CSS 大小	482	20.5 小结	522
18.2.1 CSS 高宽	482	第 21 章 多线程	523
18.2.2 元素高宽	483	21.1 浏览器线程	523
18.2.3 视窗大小	484	21.2 WebWorker	524
18.3 CSS 动画	486	21.2.1 创建线程	524
18.3.1 淡入淡出	486	21.2.2 线程通信	525
18.3.2 滑入滑出	488	21.2.3 结束线程	528
18.4 小结	491	21.3 线程安全	529
18.5 知识拓展	491		

21.4 小结	529	22.6.1 绑定与解绑	579
21.5 知识拓展	530	22.6.2 事件模拟	581
第 22 章 jQuery	534	22.6.3 事件对象	582
22.1 jQuery 介绍	534	22.6.4 文档初始化	583
22.1.1 元素集合	535	22.7 数据	584
22.1.2 链式调用	536	22.7.1 获取数据	585
22.2 选择器	537	22.7.2 设置数据	586
22.2.1 CSS 选择器	537	22.7.3 移除数据	587
22.2.2 表单选择器	539	22.8 动画	588
22.2.3 元素筛选	541	22.8.1 显示隐藏	588
22.2.4 子元素筛选	542	22.8.2 滑动动画	590
22.2.5 内容筛选	544	22.8.3 淡入淡出	591
22.2.6 可见性筛选	545	22.8.4 动画队列	592
22.3 DOM	547	22.8.5 清除动画	595
22.3.1 新建节点	547	22.8.6 自定义动画	596
22.3.2 拷贝节点	548	22.9 AJAX	598
22.3.3 插入节点	550	22.9.1 发送请求	598
22.3.4 移除节点	555	22.9.2 全局事件	603
22.3.5 替换节点	558	22.10 小结	604
22.3.6 遍历节点	560	22.11 知识拓展	605
22.4 属性操作	561	22.11.1 框架兼容	605
22.4.1 获取属性	561	22.11.2 插件开发	606
22.4.2 设置属性	563	第 23 章 项目实战	610
22.4.3 移除属性	564	23.1 项目分析	610
22.5 样式表	565	23.2 技术选型	610
22.5.1 获取样式	565	23.3 个税计算器	611
22.5.2 设置样式	566	23.3.1 网页结构	611
22.5.3 单位	570	23.3.2 税率表	613
22.5.4 尺寸	571	23.3.3 五险一金	613
22.5.5 位置	574	23.3.4 纳税计算	614
22.6 事件	579	23.4 小结	624

» 第 1 章

走进JavaScript <<

在互联网上，JavaScript用于实现五彩缤纷的网页，是网页编程的主流语言。它不需要编译，简单、易学易用，直接嵌入在网页中，让网页具有与用户交互的能力。本章将介绍JavaScript的起源和特点、如何配置开发环境，并提供学好JavaScript的建议，让读者能快速了解这门语言。

1.1 JavaScript简介

JavaScript是Web编程中一种常见的解释执行的脚本类编程语言。它主要是嵌入HTML代码中，由浏览器或WebView逐行解释执行。它通过响应各种事件，如键盘按键、鼠标拖拽、手指触摸等，来与外界交互，从而实现各种功能。

JavaScript主要在浏览器上运行，因此它具有跨平台的特性，但同时也受到浏览器的限制。因为各个浏览器厂商根据语言标准实现的JavaScript运行环境并不完全相同，所以在特定情况下，开发人员需要根据不同浏览器的差异编写不同的代码，以实现相同的功能。

随着技术发展，JavaScript不仅可以在前端浏览器运行，也可以在后端服务器上运行。目前流行的JavaScript后端运行环境Node.js支持使用JavaScript连接数据库、提供Restful接口。在Node.js出现之前，这些功能只能依赖Java、PHP等后端语言实现。JavaScript也能够与APP结合，通过APP提供的接口丰富APP的功能，比如轻应用、微信小程序、支付宝小程序等。

JavaScript应用广泛，涉及网页端程序、服务器端程序、手机APP开发等。本书将讲解JavaScript的语法基础，并主要呈现网页上的JavaScript应用。在讲解的过程中，本书会提供相应的简单易用的JavaScript代码，让读者体会到编程的乐趣。在熟练掌握该语言基础之后，相信读者也能够很容易地进行网页编程。

万丈高楼平地起，让我们开始JavaScript精彩的学习之旅吧！

1.1.1 JavaScript的起源

早期的万维网网页功能简单，大部分功能依赖后端语言实现。例如：表单验证需要将数据传输到服务器，等待服务器返回验证结果。当网速较慢时，在点击提交表单按钮、等待服务器处理数据的期间，网页没有任何有用的提示，因此用户的体验很差。

1995年，太阳计算机系统有限公司（Sun Microsystems，简称Sun公司）发布了Java和HotJava浏览器，率先推出能嵌套在网页上并跟随网络传输的Applet网页小程序。用户访问网页时，浏览器下载并运行Applet小程序，实现简单的编程交互。这种实现方式比较笨重，网景通信公司（Netscape）开始考虑开发一种浏览器直接支持运行的脚本语言。同年，网景通信公司的布兰登·艾奇（Brendan Eich）开发设计了LiveScript，并嵌入到当年准备发布的网景导航者（Netscape Navigator）浏览器上。随后，网景通信公司与Sun公司成立了一个开发联盟，关系由竞争变为合作。在发布前，网景希望搭上媒体热炒Java的顺风车，让这门新语言看起来像Java，所以临时将LiveScript改名为JavaScript。

JavaScript 1.0版本发布之后大受欢迎，获得了巨大成功，网景公司随即在Netscape Navigator 3中发布了JavaScript 1.1版本。

作为竞争对手，微软紧随其后，在Internet Explorer 3中发布了一个JavaScript的克隆版本——JScript。微软发布JScript，既可以避免与网景的潜在纠纷，如商标、专利等，又可以在Windows操作系统上捆绑销售浏览器。

提示

JavaScript与Java没有任何关系，所属领域也不同；JavaScript得到了浏览器厂商的支持，发展良好，而Java Applet已难觅踪影。

除了JavaScript、JScript外，早期还有一种浏览器嵌入式脚本语言ScriptEase，这三者没有统一的语言标准。随着不同版本的发布，语言标准混乱的问题日益明显，JavaScript的规范化被提上日程。

1997年，JavaScript 1.1草案被提交给了欧洲计算机制造商协会（ECMA）。同年，ECMA-262标准发布，命名了全新的脚本语言ECMAScript，全称European Computer Manufacturers Association Script。ECMAScript和JavaScript的区别是，前者是一套语言标准，后者是该标准的实现形式。

1.1.2 JavaScript的特点

JavaScript作为一门解释执行的脚本语言，具有如下几个特点。

1. 简单性

JavaScript语法简单，流程控制与大部分语言类似，变量没有严格的类型区分，函数没有返回类型，易于学习、理解和上手。

2. 解释性

JavaScript源代码不需要经过编译，可以直接在浏览器中解释执行。在解释执行前，浏览器需要进行预处理，包括语法检测、运行环境设置、变量提升等。

3. 动态性

JavaScript以原型为模板创建实例，实例和原型之间具有一种松散的连接关系。JavaScript也可