HZ BOOKS
华章科技

从基本语法、应用架构、工具框架、编码风格、编程思想5个方面深入探讨
编写高质量JavaScript代码的技巧、禁忌和最佳实践

實戰

成林 著

Writing Solid JavaScript Code 188 Suggestions to Improve Your JavaScript Program

编写高质量代码

改善JavaScript程序的188个建议

机械工业出版社
China Machine Press

實戰

Writing Solid JavaScript Code: 188 Suggestions to Improve Your JavaScript Program

编写高质量代码

改善JavaScript程序的188个建议

成林 著

机械工业出版社
China Machine Press

本书是 Web 前端工程师进阶修炼的必读之作，将为你通往“JavaScript 技术殿堂”指点迷津！内容全部由编写高质量的 JavaScript 代码的最佳实践组成，从基本语法、应用架构、工具框架、编码风格、编程思想等 5 大方面对 Web 前端工程师遇到的疑难问题给出了经验性的解决方案，为 Web 前端工程师如何编写更高质量的 JavaScript 代码提供了 188 条极为宝贵的建议。对于每一个问题，不仅以建议的方式给出了被实践证明为十分优秀的解决方案，而且还给出了经常被误用或被错误理解的不好的解决方案，从正反两个方面进行了分析和对比，犹如醍醐灌顶，让人豁然开朗。

本书针对每个问题所设计的应用场景都非常典型，给出的建议也都与实践紧密结合。书中的每一条建议都可能在你的下一行代码、下一个应用或下一个项目中被用到，建议你将此书放置在手边，随时查阅，一定能使你的学习和开发工作事半功倍。

封底无防伪标均为盗版
版权所有，侵权必究
本书法律顾问 北京市展达律师事务所

图书在版编目（CIP）数据

编写高质量代码：改善 JavaScript 程序的 188 个建议 / 成林著 . —北京：机械工业出版社，2012.11

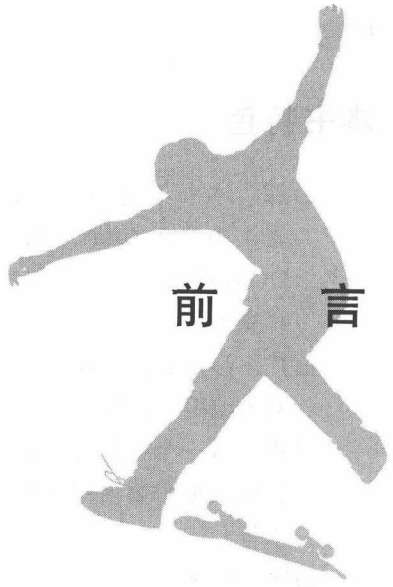
ISBN 978-7-111-39905-6

I. 编… II. 成… III. JAVA 语言 – 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字（2012）第 231084 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）
责任编辑：马 超
北京市荣盛彩色印刷有限公司印刷
2013 年 1 月第 1 版第 1 次印刷
186mm×240mm • 25.5 印张
标准书号：ISBN 978-7-111-39905-6
定价：69.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991；88361066
购书热线：（010）68326294；88379649；68995259
投稿热线：（010）88379604
读者信箱：hzjsj@hzbook.com



前　言

为什么要写这本书

JavaScript 是目前比较流行的 Web 开发语言。随着移动互联网、云计算、Web 3.0 和客户端开发概念的升温，JavaScript 语言不断成熟和普及，并被广泛应用于各种 B/S 架构的项目和不同类型的网站中。对于 JavaScript 初学者、网页设计爱好者以及 Web 应用开发者来说，熟练掌握 JavaScript 语言是必需的。

JavaScript 语言的最大优势在于灵活性好，适应能力强。借助各种扩展技术、开源库或框架，JavaScript 能够完成 Web 开发中各种复杂的任务，提升客户端用户体验。

作为资深的 Web 开发人员，笔者已经习惯了与高性能的编程语言和硬件打交道，因此一开始并没有对 JavaScript 编程有太高的期望。后来才发现，JavaScript 实际上是一种优秀且高效的编程语言，而且随着浏览器对其更好的支持、JavaScript 语言本身的性能提升，以及新的工具库加入，JavaScript 不断变得更好。JavaScript 结合 HTML 5 等为 Web 开发人员提供了真正可以发挥想象力的空间。Node.js 等新技术则为使用 JavaScript 对服务器进行编程描绘了非常美好的未来。

但是，在阅读网上大量散存的 JavaScript 代码时，笔者能明显感觉到很多用户正在误入“歧途”：编写的代码逻辑不清，结构混乱，缺乏编程人员应有的基本素养。这种现状一般都是用户轻视 JavaScript 语言所致。还有很多用户属于“半路出家”，误认为 JavaScript 就是一种“玩具语言”，没有以认真的态度对待和学习这门语言，书写代码也很随意。因此，笔者萌生了写一本以提高 JavaScript 代码编写质量为目的的书籍，在机械工业出版社华章公司杨福川编辑的鼓励和指导下，经过近半年的策划和准备，终于鼓起勇气动笔了。

本书特色

- **深**。本书不是一本语法书，它不会教读者怎么编写 JavaScript 代码，但它会告诉读者，为什么 Array 会比 String 类型效率高，闭包的自增是如何实现的，为什么要避免 DOM 迭代……不仅仅告诉读者 How（怎么做），而且还告诉读者 Why（为什么要这样做）。
- **广**。涉及面广。从编码规则到编程思想，从基本语法到系统框架，从函数式编程到面向对象编程，都有涉及，而且所有的建议都不是“纸上谈兵”，都与真实的场景相结合。
- **点**。从一个知识点展开讲解，比如继承，这里不提供继承的解决方案，而是告诉读者如何根据需要使用继承，如何设置原型，什么时候该用类继承，什么时候该用原型继承等。
- **精**。简明扼要。一个建议就是对一个问题的解释和说明，以及相关的解决方案，不拖泥带水，只针对一个问题进行讲解。
- **洁**。虽然笔者尽力把每个知识点写得生动，但代码就是代码，很多时候容不得深加工，最直接也就是最简洁的。

这是一本建议书。有这样一本书籍在手边，对如何编写出优雅而高效的代码提供指导，将是一件多么惬意的事情啊！

读者对象

本书适合以下读者阅读：

- 打算学习 JavaScript 的开发人员。
- 有意提升自己网站水平和 Web 应用程序开发能力的 Web 开发人员。
- 希望全面深入理解 JavaScript 语言的初学者。

此外，本书也适合熟悉下列相关技术的读者阅读：

- PHP/ASP/JSP
- HTML/ XML
- CSS

对于没有计算机基础知识的初学者，以及只想为网站添加简单特效和交互功能的读者，阅读本书前建议先阅读 JavaScript 基础教程类图书。

如何阅读本书

本书将改善 JavaScript 编程质量的 188 个建议以 9 章内容呈现：

此为试读，需要完整PDF请访问：www.ertongbook.com

❑第 1 章　JavaScript 语言基础

JavaScript 中存在大量的问题，这些问题会妨碍读者编写优秀的程序。应该避免 JavaScript 中那些糟糕的用法，因此本章主要就 JavaScript 语言的一些基本用法中容易犯错误的地方进行说明，希望能够引起读者的重视。

❑第 2 章　字符串、正则表达式和数组

JavaScript 程序与字符串操作紧密相连，在进行字符串处理时无时无刻不需要正则表达式的帮忙。如何提高字符串操作和正则表达式运行效率是很多开发者最易忽视的问题。同时，数组是所有数据序列中运算速度最快的一种类型，但很多初学者忽略了这个有用的工具。本章将就这 3 个技术话题展开讨论，通过阅读这些内容相信读者能够提高程序的执行效率。

❑第 3 章　函数式编程

函数式编程已经在实际应用中发挥了巨大作用，越来越多的语言不断地加入对诸如闭包、匿名函数等的支持。从某种程度上来讲，函数式编程正在逐步同化命令式编程。当然，用好函数并非易事，需要“吃透”函数式编程的本质，本章帮助读者解决在函数式编程中遇到的各种问题。

❑第 4 章　面向对象编程

JavaScript 采用的是以对象为基础，以函数为模型，以原型为继承机制的开发模式。因此，对于习惯于面向对象开发的用户来说，需要适应 JavaScript 语言的灵活性和特殊性。本章将就 JavaScript 类、对象、继承等抽象的问题进行探索，帮助读者走出“误区”。

❑第 5 章　DOM 编程

DOM 操作代价较高，在富网页应用中通常是一个性能瓶颈。因此，在 Web 开发中，需要特别注意性能问题，尽可能地降低性能损耗。本章将为读者提供一些好的建议，帮助读者优化自己的代码，让程序运行得更快。

❑第 6 章　客户端编程

在 JavaScript 开发中，很多交互效果都需要 CSS 的配合才能够实现，因此 CSS 的作用不容忽视。本章主要介绍 JavaScript+CSS 脚本化编程，以及 JavaScript 事件控制技巧。

❑第 7 章　数据交互和存储

数据交互和存储是 Web 开发中最重要的，也是最容易被忽视的问题，它也是高性能 JavaScript 的基石，是提升网站可用性的最大要素。本章主要介绍如何使用 JavaScript 提升数据交互的反应速度，以便更好地让数据在前、后台传递。

❑第 8 章　JavaScript 引擎与兼容性

JavaScript 兼容性是 Web 开发的一个重要问题。为了实现浏览器解析的一致性，需要找出不同引擎的分歧点在哪里。本章主要介绍各主流引擎在解析 JavaScript 代码时的分歧，使读者能够编写出兼容性很高的代码。

❑ 第 9 章　JavaScript 编程规范和应用

每种语言都存在缺陷。事实证明代码风格在编程中是非常重要的，好的风格促使代码能被更好地阅读，更为关键的是，它能够提高代码的执行效率。本章主要介绍如何提升 JavaScript 代码编写水平，主要包括风格、习惯、效率、协同性等问题，希望能够给读者带来帮助。

本书的期望

您是否曾经为了提供一个简单的应用解决方案而彻夜地查看源代码？

您是否曾经为了理解某个框架而冥思苦想、阅览群书？

您是否曾经为了提升 0.1s 的 DOM 性能而对多种实现方案进行严格测试和对比？

您是否曾经为了避免兼容问题而遍寻高手共同“诊治”？

……

在学习和使用 JavaScript 的过程中，您是否在原本可以很快掌握或解决的问题上耗费了大量的时间和精力？本书的很多内容都是笔者曾经付出代价换来的，希望它们能够给您带来一些帮助！

代码是一切的基石，一切都是以编码实现为前提的，通过阅读本书，期望为读者带来如下帮助：

❑ 能写出简单、清晰、高效的代码。

❑ 能架构一个稳定、健壮、快捷的应用框架。

❑ 能回答一个困扰很多人的技术问题。

❑ 能修复一个应用开发中遇到的大的 Bug。

❑ 能非常熟悉某个开源产品。

❑ 能提升客户端应用性能。

……

但是，“工欲善其事，必先利其器”，在“善其事”之前，先检查“器”是否已经磨得足够锋利了，是否能够在前进的路上披荆斩棘。无论将来的职业发展方向是架构师、设计师、分析师、管理者，还是其他职位，只要还与软件打交道，就有必要打好技术基础。本书所涉及的全部是核心的 JavaScript 编程技术，如果能全部理解并付诸实践，一定可以夯实 JavaScript 编程基础。

勘误和支持

除封面署名外，对本书编写提供帮助的还有：马本连、吴建华、江淑军、李斌、李经键、

郑伟、田蜜、陆颖、王慧明、张炜、陈锐、王幼平、杨龙贵、苏震巍、崔鹏飞等。由于作者的水平有限，加之编写时间仓促，书中难免会出现一些错误或不准确的地方，恳请读者批评指正。书中的全部源文件可以从华章网站（www.hzbook.com）下载。如果您有任何意见建议，欢迎发送邮件至邮箱 js_code@126.com，期待得到您的真挚反馈。

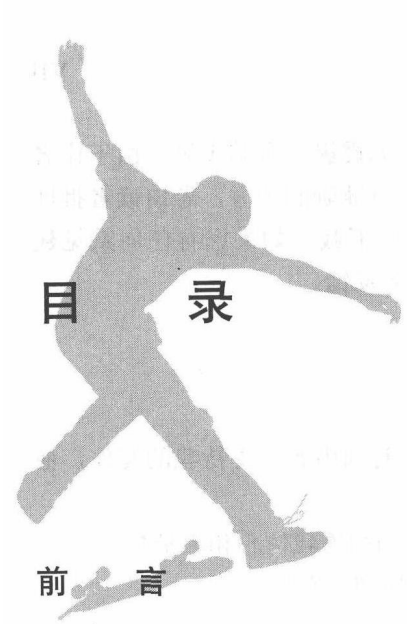
致谢

感谢机械工业出版社华章公司的杨福川编辑在这一年多的时间中始终支持我的写作，他的鼓励和帮助让我顺利完成了本书的编写工作。

最后感谢我的父母，感谢他们的养育之恩，感谢他们时时刻刻给我信心和力量！

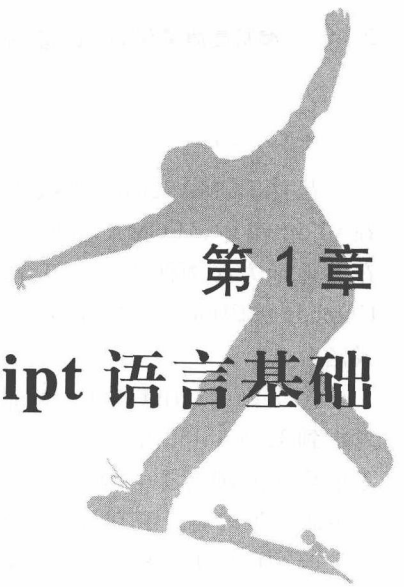
谨以此书献给我最亲爱的家人，以及众多热爱 JavaScript 的朋友们！

成林



目　录

此为试读，需要完整PDF请访问：www.ertongbook.com



第 1 章
JavaScript 语言基础

对于任何语言来说，如何选用代码的写法和算法最终会影响到执行效率。与其他语言不同，由于 JavaScript 可用资源有限，所以规范和优化更为重要。代码结构是执行速度的决定因素之一：代码量少，运行速度不一定快；代码量多，运行速度也不一定慢。性能损失与代码的组织方式及具体问题的解决办法直接相关。

程序通常由很多部分组成，具体表现为函数、语句和表达式，它们必须准确无误地按照顺序排列。优秀的程序应该拥有前瞻性的结构，可以预见到未来所需要的修改。优秀的程序也有一种清晰的表达方式。如果一个程序被表达得很好，那么它更容易被理解，进而能够成功地被修改或修复。JavaScript 代码经常被直接发布，因此它应该自始至终具备发布质量。整洁是会带来价值的，通过在一个清晰且始终如一的风格下编写的程序会更易于阅读。

JavaScript 的弱类型和过度宽容特征，没有为程序质量带来安全编译时的保证，为了弥补这一点，我们应该按严格的规范进行编码。JavaScript 包含大量脆弱的或有问题的特性，这些会妨碍编写优秀的程序。我们应该避免 JavaScript 中那些糟糕的特性，还应该避免那些通常很有用但偶尔有害的特性。这样的特性让人既爱又恨，避免它们就能避免日后开发中潜在的错误。

建议 1：警惕 Unicode 乱码

ECMA 标准规定 JavaScript 语言基于 Unicode 标准进行开发，JavaScript 内核完全采用 UCS 字符集进行编写，因此在 JavaScript 代码中每个字符都使用两个字节来表示，这意味着可以使用中文来命名变量或函数名，例如：

```
var 人名 = "张三";
function 睡觉(谁){
    alert(谁 + ": 快睡觉！都半夜三更了。");
```

```
}
睡觉（人名）;
```

虽然 ECMAScript v3 标准允许 Unicode 字符出现在 JavaScript 程序的任何地方，但是在 v1 和 v2 中，ECMA 标准只允许 Unicode 字符出现在注释或引号包含的字符串直接量中，在其他地方必须使用 ASCII 字符集，在 ECMAScript 标准化之前，JavaScript 通常是不支持 Unicode 编码的。考虑到 JavaScript 版本的兼容性及开发习惯，不建议使用汉字来命名变量或函数名。

由于 JavaScript 脚本一般都“寄宿”在网页中，并最终由浏览器来解析和执行，因此在考虑到 JavaScript 语言编码的同时，还要顾及嵌入页面的字符编码，以及浏览器支持的编码。不过现在的浏览器一般都支持不同类型的字符集，只需要考虑页面字符编码与 JavaScript 语言编码一致即可，否则就会出现乱码现象。

当初设计 JavaScript 时，预计最多会有 65 536 个字符，从那以后慢慢增长到了一百万个字符。JavaScript 字符是 16 位的，这足够覆盖原有的 65 536 个字符，剩下的百万字符中的每一个都可以用一对字符来表示。

Unicode 把一对字符视为一个单一的字符，而 JavaScript 认为一对字符是两个不同的字符，这将会带来很多问题，考虑到代码的安全性，我们应该尽量使用基本字符进行编码。

建议 2：正确辨析 JavaScript 句法中的词、句和段

JavaScript 语法包含了合法的 JavaScript 代码的所有规则和特征，它主要分为词法和句法。词法包括字符编码、名词规则、特殊词规则等。词法侧重语言的底层实现（如语言编码问题等），以及基本规则的定义（如标识符、关键字、注释等）。它们都不是最小的语义单位，却是构成语义单位的组成要素。例如，规范字符编码集合、命名规则、标识符、关键字、注释规则、特殊字符用法等。

句法定义了语言的逻辑和结构，包括词、句和段的语法特性，其中段体现逻辑的结构，句表达可执行的命令，词演绎逻辑的精髓。

段落使用完整的结构封装独立的逻辑。在 JavaScript 程序中，常用大括号来划分结构，大括号拥有封装代码和逻辑的功能，由此形成一个独立的段落结构。例如，下面这些结构都可以形成独立的段落。

```
{
    // 对象
}
function (){
    // 函数
}
if (){
    // 条件
```

此为试读，需要完整PDF请访问：www.ertongbook.com