<<编写高质量代码>>

图书基本信息

书名：<<编写高质量代码>>

13位ISBN编号：9787111399056

10位ISBN编号：7111399056

出版时间：2012-11

出版时间：机械工业出版社

作者：成林

页数：393

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、Web3.0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/ XMLCSS 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：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。

<<编写高质量代码>>

内容概要

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。

作者简介

成林，资深Web前端工程师，从事Web前端工作多年，精通CSS、HTML、JavaScript、jQuery和Ajax等Web前端技术，在实践中积累了大量的经验。
推崇Web技术标准，曾经在多所高等院校和一些线下技术沙龙主讲Web标准和规范相关的课程，曾经还参与过W3C组织的标准化文档的中文编译工作。
近几年来，集中精力研究和实践CSS
3和 HTML 5前沿技术，在国内是该领域的先驱者之一。

<<编写高质量代码>>

书籍目录

章节摘录

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。

编辑推荐

《编写高质量代码:改善JavaScript程序的188个建议》编辑推荐：从语法、编程思想、编码规范、工具方法总结出188个编写高质量JS代码的技巧、禁忌和最佳实践！

版权说明