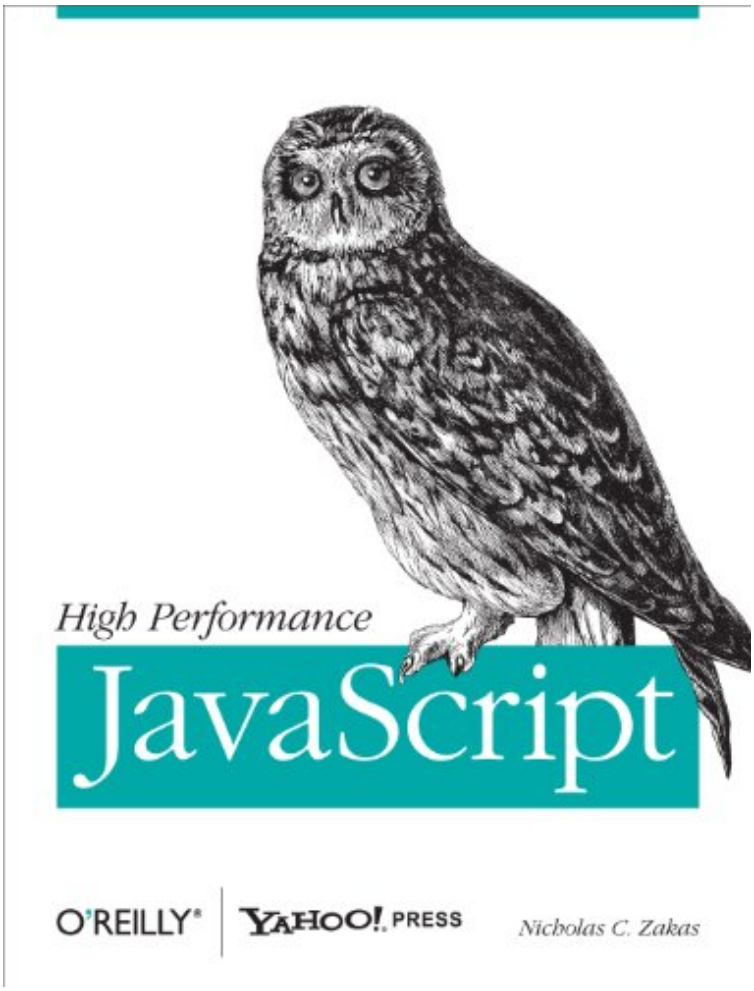
High Performance JavaScript: Build Faster Web Application Interfaces

Von Nicholas C. Zakas
*ebooks | Download PDF | *ePub | DOC | audiobook*



Produktinformation -Verkaufsrang: #37394 in eBooksVerffentlicht am: 2010-03-11Erscheinungsdatum: 2010-03-11File Name: B0043D2F62 | File size: 43.Mb

Von Nicholas C. Zakas : High Performance JavaScript: Build Faster Web Application Interfaces before purchasing it in order to gage whether or not it would be worth my time, and all praised High Performance JavaScript: Build Faster Web Application Interfaces:

KundenrezensionenHilfreichste Kundenrezensionen1 von 1 Kunden fanden die folgende Rezension hilfreich. Interessante LektreVon Selda B.Klasse Buch. Viele Aspekte von JavaScript und JavaScript im Browser werden Klasse erklrt. Klar findet man vieles inzwischen auch im Netz kostenlos, aber Ich finde ein Nachschlagewerk in Buchform immer besser.Die Beispiele sind Verstndlich und die Tools die im Buch vorgestellt werden helfen einem Entwickler

sehr.

KurzbeschreibungIf you're like most developers, you rely heavily on JavaScript to build interactive and quick-responding web applications. The problem is that all of those lines of JavaScript code can slow down your apps. This book reveals techniques and strategies to help you eliminate performance bottlenecks during development. You'll learn how to improve execution time, downloading, interaction with the DOM, page life cycle, and more.Yahoo! frontend engineer Nicholas C. Zakas and five other JavaScript expertsRoss Harmes, Julien Lecomte, Steven Levithan, Stoyan Stefanov, and Matt Sweeneydemonstrate optimal ways to load code onto a page, and offer programming tips to help your JavaScript run as efficiently and quickly as possible. You'll learn the best practices to build and deploy your files to a production environment, and tools that can help you find problems once your site goes live.Identify problem code and use faster alternatives to accomplish the same taskImprove scripts by learning how JavaScript stores and accesses dataImplement JavaScript code so that it doesn't slow down interaction with the DOMUse optimization techniques to improve runtime performanceLearn ways to ensure the UI is responsive at all timesAchieve faster client-server communicationUse a build system to minify files, and HTTP compression to deliver them to the browser