

Programmeren met JavaScript voor dummies®

SPIEKBRIEF

Leer programmeren met JavaScript door te ontdekken welke sleutelwoorden er gebruikt worden in JavaScript inclusief een uitgebreide lijst met HTML5 API's. Kijk om te beginnen eens naar deze handige tips.

Sleutelwoorden voor JavaScript

Wanneer je programmeert met JavaScript, is het handig als je bepaalde sleutelwoorden kent. In de volgende lijst staan de sleutelwoorden voor JavaScript. Je kunt deze woorden niet gebruiken als JavaScript-variabelen, functies, methoden, luslabels of objectnamen.

abstract	final	public
boolean	finally	return
break	float	short
byte	for	static
case	function	super
catch	goto	switch
char	if	synchronized
class	implements	this
const	import	throw
continue	in	throws
debugger	instanceof	transient
default	int	true
delete	interface	try
do	long	typeof
double	native	var
else	new	void
enum	null	volatile
export	package	while
extends	private	with
false	protected	

Programmeren met JavaScript ^{voor} dummies®

SPIEKBRIEF

Lijst met HTML5 API's om te programmeren met JavaScript

Hierna volgt een uitgebreide lijst met HTML5 API's die zijn voorgesteld of geïmplementeerd. Browsers worden continu bijgewerkt met nieuwe functies. Ga naar caniuse.com voor het laatste nieuws over welke browsers welke functies ondersteunen.

API	Wat deze doet
Ambient Light API	Geeft informatie over de omgevingslichtniveaus zoals gedetecteerd door de lichtsensor van een apparaat.
Battery Status API	Geeft informatie over de accustatus van een apparaat.
Canvas 2D Context	Maakt tekenen en bewerken van afbeeldingen in een browser mogelijk.
Clipboard API	Geeft toegang tot de kopieer-, knip- en plakfunctionaliteit van het besturingssysteem.
Contacts	Geeft toegang tot de opslagplaats van de contacten van een gebruiker in een webbrowser.
Drag and Drop	Ondersteunt het slepen en neerzetten van items binnen en tussen browservensters.
File API	Biedt programma's veilige toegang tot het bestandssysteem van het apparaat.
Forms	Biedt programma's toegang tot de nieuwe gegevenstypen die zijn gedefinieerd in HTML5.
Fullscreen API	Stuurt het gebruik van het volledige scherm aan van de gebruiker voor webpagina's, zonder de browser-UI.
Gamepad API	Ondersteunt invoer van USB-gamepadcontrollers.
Geolocation	Geeft webprogramma's toegang tot geografische locatiegegevens over het apparaat van de gebruiker.
getUserMedia/Stream API	Biedt toegang tot gegevens over externe apparaten (zoals webcamvideo).
History API	Stelt programma's in staat om de browsergeschiedenis te bewerken.
HTML Microdata	Voorziet in een manier om inhoud te annoteren met door computer leesbare labels.
Indexed database	Creëert een eenvoudig databasesysteem aan cliëntzijde in de webbrowser.
Internationalization API	Biedt toegang tot landinstellingsgevoelige opmaak en stringvergelijking.
Offline apps	Hiermee kunnen programmeurs webapps beschikbaar maken in offline modus.
Proximity API	Geeft informatie over de afstand tussen een apparaat en een object.
Screen Orientation	Leest de schermstand (staand of liggend) en geeft programmeurs de mogelijkheid om te weten wanneer deze verandert en om de stand te vergrendelen.
Selection	Ondersteunt het selecteren van elementen in JavaScript met behulp van CSS-stijlselectors.
Server-sent events	Hiermee kan de server gegevens pushen naar de browser zonder dat de browser erom hoeft te vragen.
User Timing API	Geeft programmeurs toegang tot tijdstempels met hoge nauwkeurigheid om de prestatie van programma's te meten.
Vibration API	Biedt toegang tot de trilfunctionaliteit van het apparaat.
Web Audio API	API om audio te verwerken en synthetiseren.
Web Messaging	Hiermee kunnen browservensters van verschillende oorsprong met elkaar communiceren.
Web Speech API	Biedt functies voor spraakinvoer en tekst-naar-spraakuitvoer.
Web storage	Maakt het mogelijk om sleutel-waardeparen op te slaan in de browser.
Web sockets	Opent een interactieve communicatiesessie tussen de browser en server.
Web Workers	Hiermee kan JavaScript scripts op de achtergrond uitvoeren.
XMLHttpRequest2	Verbeterd XMLHttpRequest zodat SOP-fouten niet meer hoeven te worden omzeild en om XMLHttpRequest te laten werken met nieuwe functies van HTML5.

Inhoud in vogelvlucht

Inleiding	1
Deel 1: Aan de slag met JavaScript	5
HOOFDSTUK 1: De meest onbegrepen programmeertaal ter wereld	7
HOOFDSTUK 2: Je eerste JavaScript-programma schrijven	19
HOOFDSTUK 3: Werken met variabelen	41
HOOFDSTUK 4: Arrays begrijpen	59
HOOFDSTUK 5: Werken met operatoren, expressies en opdrachten	71
HOOFDSTUK 6: Meegaan met de flow met lussen en vertakkingen	87
Deel 2: Je JavaScript organiseren	101
HOOFDSTUK 7: Functioneel aan de slag	103
HOOFDSTUK 8: Objecten maken en gebruiken	123
Deel 3: JavaScript op het web	139
HOOFDSTUK 9: De browser aansturen met het object Window	141
HOOFDSTUK 10: Documenten manipuleren met het DOM	155
HOOFDSTUK 11: Events gebruiken in JavaScript	177
HOOFDSTUK 12: Invoer en uitvoer integreren	189
HOOFDSTUK 13: Werken met CSS en afbeeldingen	205
Deel 4: Voorbij de basis	223
HOOFDSTUK 14: Zoeken met reguliere expressies	225
HOOFDSTUK 15: Callbacks en closures	239
HOOFDSTUK 16: AJAX en JSON omarmen	251
Deel 5: JavaScript en HTML5	267
HOOFDSTUK 17: HTML5 API's	269
HOOFDSTUK 18: jQuery	285
Deel 6: Het deel van de tientallen	303
HOOFDSTUK 19: Tien JavaScript-frameworks en -bibliotheken	305
HOOFDSTUK 20: Tien veelvoorkomende JavaScript-bugs en hoe je ze vermijdt	317
HOOFDSTUK 21: Tien online tools om je te helpen beter JavaScript te schrijven	327
Index	335

Inleiding

JavaScript is hot! Wat begon als een snelle taal gemaakt voor een van de eerste webbrowsers is nu een van de populairste programmeertalen ter wereld. De vraag naar JavaScript-programmeurs is op een ongekend hoog punt en blijft alleen maar toenemen.

Dit boek is jouw sleutel om een expert te worden in de kernconcepten van JavaScript. Of je nu een goed betaalde baan als programmeur wilt of je eigen persoonlijke website interactiever wilt maken, de informatie en technieken die in dit boek staan, zijn volledig up-to-date met de actueelste standaarden en best practices van JavaScript.

Aan elk hoofdstuk zijn leuke en interactieve online oefeningen gekoppeld. Verder bevat elk hoofdstuk complete voorbeelden van echte code die je kunt proberen en thuis in je eigen webbrowser kunt testen.

Net zoals je Carnegie Hall alleen kunt bereiken door te oefenen, oefenen en nog eens oefenen, kun je alleen een betere programmeur worden door te programmeren, programmeren en nog eens te programmeren!

Over dit boek

Dit boek is een vriendelijke en toegankelijke handleiding om te beginnen met het schrijven van JavaScript-code. Wat programmeertalen betreft, is JavaScript vrij eenvoudig te leren en gebruiken. Omdat de taal zo toegankelijk is, bevinden veel mensen die zijn begonnen als auteurs van webpagina's zich nu in de positie dat ze verantwoordelijk zijn voor het onderhouden, wijzigen en schrijven van JavaScript-code. Als jij je door deze beschrijving voelt aangesproken, kun je je kennis snel en eenvoudig bijspijkeren aan de hand van dit boek.

Of je nu al een beetje afweet van JavaScript of het nog nooit eerder bent tegengekomen, dit boek laat je zien hoe je op de juiste manier JavaScript schrijft.

In dit boek worden onder andere de volgende onderwerpen behandeld:

- » de basisstructuren van JavaScript-programma's;
- » JavaScript integreren met HTML5 en CSS3;
- » programma's structureren met functies;
- » werken met JavaScript-objecten;
- » geavanceerde JavaScript-technieken gebruiken, zoals AJAX, callbacks en closures;
- » aan de slag met jQuery.

Om JavaScript te leren, moet je niet alleen de syntaxis van de taal leren. Je moet ook iets afweten van de tools en de gemeenschap die zich heeft gevormd rondom de taal. In de loop van de lange geschiedenis van JavaScript hebben professionele JavaScript-programmeurs de tools en technieken om JavaScript te schrijven enorm verbeterd. In dit boek noemen we belangrijke best practices en tools om sneller betere code te testen, documenteren en schrijven!

Houd het volgende in je achterhoofd wanneer je dit boek leest:

- » Alle JavaScript-code en alle HTML- en CSS-mark-up zijn in monospace gezet:

```
document.write("Hi!");
```

- » In de marges van een boekpagina is niet zo veel ruimte als op je beeldscherm. Lange regels met HTML, CSS en JavaScript kunnen daarom worden afgebroken en op meerdere regels komen te staan. Onthoud dat je computer deze regels wel beschouwt als enkele regels HTML, CSS of JavaScript. We geven aan dat alles op één regel moet staan door de regel af te breken op een interpunctieteken of een spatie en dan eventuele resterende tekst in te springen:

```
document.getElementById("anElementInTheDocument").  
addEventListener("click", doSomething, false);
```

- » Het maakt HTML en CSS niet zo veel uit of je nu hoofdletters of kleine letters of een combinatie daarvan gebruikt, maar JavaScript wel! Om ervoor te zorgen dat je de correcte resultaten krijgt van de codevoorbeelden in het boek, moet je altijd dezelfde hoofdletters en kleine letters gebruiken die wij ook gebruiken.

Dwaze veronderstellingen

Bij ons bedrijf, WatzThis?, volgen we de stelregel om nooit iets te veronderstellen (maar eerlijk gezegd is Eva hier beter in dan Chris).

Je hoeft geen programmeerinja of een hacker te zijn om programmeren te begrijpen. Je hoeft niet te weten hoe de interne onderdelen van je computer werken. Je hoeft zelfs niet te weten hoe je binair telt.

We moeten echter wel een paar dingen veronderstellen over jou. We veronderstellen dat je je computer kunt aanzetten, dat je een muis en een toetsenbord kunt gebruiken en dat je een werkende internetverbinding en webbrowser hebt. Als je al een klein beetje weet hoe je webpagina's maakt (dat is echt niet moeilijk!), heb je een voorsprong.

De andere dingen die je moet weten om JavaScript-code te schrijven en uit te voeren, zijn details die we in dit boek behandelen. En nog een ding: programmeren vereist aandacht voor detail.

Pictogrammen in dit boek

Hierna volgt een lijst met de pictogrammen die we in dit boek gebruiken om aan te geven aan welke tekst en informatie je wat meer aandacht moet schenken:



TIP

Dit pictogram wijst je op handige tips die je eenvoudige of kortere manieren laten zien die je tijd of moeite besparen.



BELANGRIJK

Wanneer je dit pictogram ziet, moet je goed opletten. De informatie die hier staat, moet je zeker onthouden.



PAS OP

Wees voorzichtig, heel voorzichtig. Dit pictogram waarschuwt je voor valkuilen die je moet vermijden.



VIND
ONLINE

Dit pictogram wijst je op de fantastische oefeningen die je op de (Engelstalige) website vindt. Als je graag aan de slag wilt met JavaScript, ga je online en bezoek je www.codecademy.com/en/tracks/coding-with-javascript-for-dummies.



Dit pictogram wijst je op technische details die je misschien wel (of niet) interessant vindt. Het staat je vrij om deze informatie over te slaan, maar als je een technerd bent, vind je het misschien wel leuk.

Naast het boek

Hierna volgt een lijstje met waar je de online inhoud van dit boek vindt:

- » **Oefeningen.** Je kunt alle oefeningen online vinden op www.codecademy.com/en/tracks/coding-with-javascript-for-dummies. Daar kun je de oefeningen bij de Codecademy maken.
- » **Voorbeelden.** Je vindt alle listings met code online op www.dummies.nl/downloads. Hier vind je een zip-bestand met daarin html-bestanden met listings per hoofdstuk.

En nu?

Programmeren met JavaScript is leuk en als je er eenmaal wat van afweet, ligt de wereld van interactieve webtoepassingen aan je voeten! We hopen dat je plezier beleeft aan dit boek en onze wijze raad.

Hoofdstuk 1

De meest onbegrepen programmeertaal ter wereld

‘De mensen begrijpen me zo weinig dat ze zelfs mijn geklaag dat ze me niet begrijpen, niet begrijpen.’

– SØREN KIERKEGAARD

JavaScript heeft niet altijd zo hoog aangeschreven gestaan als vandaag de dag. Het is door enkele mensen wel de beste en de slechtste programmeertaal ter wereld genoemd. In de afgelopen paar jaar zijn er veel verbeteringen aangebracht in de manier waarop programmeurs JavaScript schrijven en in JavaScript-interpreters. Dankzij deze verbeteringen is JavaScript vandaag de dag een veel betere taal dan het ooit is geweest.

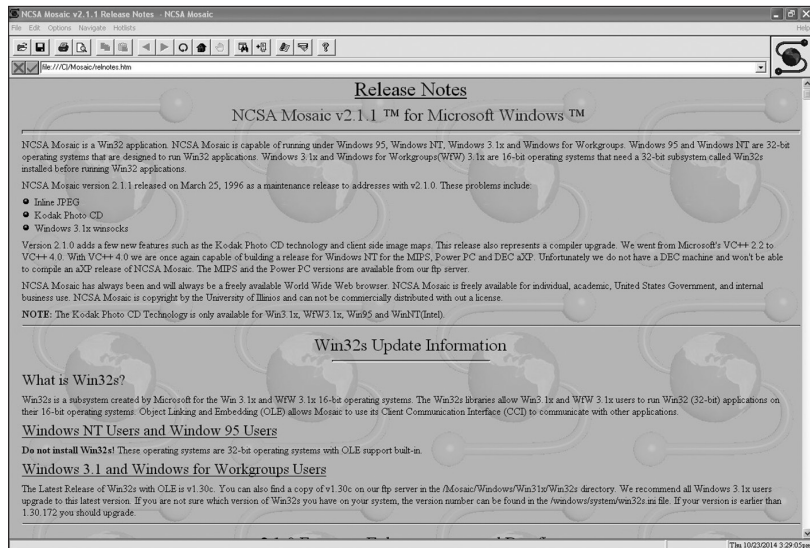
In dit hoofdstuk kom je te weten wat JavaScript is en lees je iets over de geschiedenis van de taal. Je leert ook wat JavaScript doet en waarom je de taal moet kennen.



Vergeet niet om naar de website te gaan waarop de (Engelstalige) online oefeningen staan die bij dit hoofdstuk horen!

Wat is JavaScript?

In de begindagen van het web waren browsers eenvoudige lezers voor webpagina's (zie figuur 1.1). Ze hadden zelf praktisch geen capaciteiten, behalve de mogelijkheid om tekst weer te geven in lettertypen van verschillende afmetingen. Zodra Microsoft zijn browser Internet Explorer op de markt bracht, begon de browseroorlog en vlogen de functies in het rond! Een browser introduceerde de mogelijkheid om afbeeldingen weer te geven, een andere introduceerde de mogelijkheid om verschillende lettertypen te gebruiken en daarna werden knipperende tekst, bewegende tekst en allerlei andere maffe mogelijkheden geïntroduceerd!



FIGUUR 1.1:
De eerste web-browsers stelden niet veel voor.

Het duurde niet lang voordat iemand bedacht dat browsers ook zelf handige dingen konden doen, in plaats van alleen maar te fungeren als kunstige documentweergaveprogramma's.

De Eich-mankomeet

JavaScript begon in 1995 bij Netscape. De maker van JavaScript, Brandon Eich, schreef JavaScript in recordtijd (er wordt wel gezegd in minder dan tien dagen!) door veel van de beste functies van verschillende andere

talen te lenen. Vanwege de haast om het op de markt te brengen, ontstonden enkele interessante eigenaardigheden (of iets minder beleefd gezegd, fouten) in het ontwerp van de taal. Het resultaat is een soort Esperanto-achtige taal die mensen die ervaring hebben met andere programmeertalen bedrieglijk vertrouwd voorkomt.

Mocha-licious

De oorspronkelijke naam van JavaScript was Mocha. Het kreeg de naam LiveScript met de eerste bèta-implementatie van Netscape Navigator en werd veranderd in JavaScript toen het in 1995 werd ingebouwd in de browser Netscape 2. Microsoft onderwierp JavaScript heel snel aan reverse-engineering en introduceerde een exacte kloon ervan in Internet Explorer. Ze noemden het Jscript om merkproblemen te omzeilen.

Netscape diende JavaScript in bij Ecma International, de organisatie voor normalisatie, en in 1997 werd het aangenomen en gestandaardiseerd als ECMAScript.



TECHNISCHE
INFO



PAS OP

Brandon Eich, de maker van JavaScript, maakte een beroemd geworden opmerking over de naam van de gestandaardiseerde taal en zei dat ECMAScript een ‘ongewenste merknaam is die klinkt als een huidziekte’.

ECMAScript is niet alleen een onaantrekkelijke naam voor een programmeertaal, maar de naam die Netscape aan de taal heeft gegeven en die de meeste mensen gebruiken, is ook niet heel handig. Als je al kunt programmeren in Java of als je dit op een gegeven moment gaat leren, is het een heel goed idee om in je achterhoofd te houden dat ze misschien wel enkele overeenkomsten hebben, maar dat ze verder enorm van elkaar verschillen.

We hebben meer effecten nodig!

Toen JavaScript zijn debuut maakte, werd het al heel snel heel populair als een manier om webpagina's dynamischer te maken. Zogeheten dynamische HTML (DHTML) was een van de eerste resultaten van JavaScript dat werd ingebouwd in webbrowsers en dit maakte allerlei grappige effecten mogelijk, zoals het effect van de vallende sneeuwvlokjes (zie figuur 1.2), pop-upvensters en omkrullende hoeken van webpagina's, maar ook nuttigere dingen als vervolgkeuzemenu's en formuliervalidatie.