

# Inhoud

<b>1</b>	<b>Webtalen, browsers en editors</b>	<b>1</b>
	<b>Introductie</b>	<b>2</b>
	<b>Een korte terugblik</b>	<b>2</b>
	Afspraken over webtalen	3
	Twee smaken HTML	5
	Wat HTML 5 is	6
	<b>Kennismaken met HTML en CSS</b>	<b>8</b>
	Aan de serverkant	8
	De functie van HTML	9
	De functie van CSS	10
	<b>Browsers</b>	<b>12</b>
	Google Chrome	13
	Microsoft Edge (Internet Explorer)	14
	Mozilla Firefox	15
	MacOS: Safari	16
	<b>HTML-editors</b>	<b>16</b>
	Visual Studio Code	17
	Commerciële editors/IDE's	19
	<b>Aanvullende hulpmiddelen</b>	<b>19</b>
	Hulpmiddelen voor validatie	19
	Hulpmiddelen voor ontwikkeling	20
	<b>Over dit boek</b>	<b>25</b>
	Oefenbestanden	25
	<b>Samenvatting</b>	<b>25</b>
	<b>Oefeningen</b>	<b>26</b>
<b>2</b>	<b>De basis van HTML</b>	<b>27</b>
	<b>Kenmerken van HTML-documenten</b>	<b>28</b>
	<b>Elementen, tags en tekst</b>	<b>28</b>
	Opbouw van een element	30
	Elementen nesten	31

## Inhoud

Kleine letters	32
Sluiten in de goede volgorde	32
De HTML-code van webpagina's bekijken	33
<b>Attributen</b>	<b>36</b>
Kenmerken van attributen	36
Globale attributen	37
Microdata	42
WAI-ARIA	42
<b>Kenmerken van HTML 5</b>	<b>43</b>
Omringende technieken	45
<b>De basis van een HTML 5-document</b>	<b>46</b>
<!DOCTYPE html>	47
<html lang="nl">	47
<head>	48
<meta charset="utf-8">	48
<title>	48
<link>	49
<body>	49
<b>Paginastructuur</b>	<b>50</b>
<b>Document Object Model – HTML DOM</b>	<b>50</b>
<b>Het contentmodel van HTML 5</b>	<b>51</b>
<b>Secties markeren</b>	<b>54</b>
<article> (sectioning content)	55
<section> (sectioning content)	56
<nav> (sectioning content)	57
<aside> (sectioning content)	58
Koppen: <h1> t/m <h6>	59
<header> (flow content)	60
<footer> (flow content)	61
<b>Samenvatting</b>	<b>63</b>
<b>Oefeningen</b>	<b>63</b>
<b>3 Tekst markeren</b>	<b>65</b>
<b>Inleiding</b>	<b>66</b>
<b>Koppen in HTML-documenten</b>	<b>66</b>
Het getal bepaalt de keus, niet de tekstgrootte	66
Titel website is niet automatisch <h1>	67
<b>Koppen markeren</b>	<b>67</b>
Koppen in een header	68
<b>Alineatekst markeren</b>	<b>69</b>
Alinea's: het element <p>	70
Het regeleinde  	71
Afbreken van woorden	72

<b>Speciale betekenis aangeven</b>	<b>72</b>
Klemtoon leggen met <em>	73
Nadruk leggen met <strong>	73
Aanvullende informatie is <small>	73
Ongeldig maken met <s>	73
Verwijzing naar boek of film: <cite> en <q>	73
Afkortingen en definities: <abbr> en <dfn>	74
Programmacode en dergelijke: <code>, <samp> en <kbd>	75
Subscript en superscript: <sub> en <sup>	75
Tekst markeren met <mark>	75
Het containerelement <span>	76
Toegevoegd of verwijderd: <ins> en <del>	76
<b>Inhoud groeperen</b>	<b>76</b>
<address>	76
<main>	77
Vooraf opgemaakte inhoud: <pre>	78
Citaten gebruiken: <blockquote>	79
Lijsten maken: <ol>, <ul> en <li>	79
Illustraties en bijschriften: <figure> en <figcaption>	84
FAQ, metadata en definities: <dl>, <dt>, <dd>	88
Nieuw thema markeren: <hr>	89
Het laatste redmiddel: <div>	90
<b>Speciale tekens in webpagina's</b>	<b>90</b>
Van ASCII naar UTF-8	91
Referenties gebruiken	92
<b>Samenvatting</b>	<b>92</b>
<b>Oefeningen</b>	<b>93</b>
<b>4 Koppelingen maken</b>	<b>95</b>
<b>Inleiding</b>	<b>96</b>
<b>Verbindingen leggen met &lt;a&gt;</b>	<b>97</b>
Attributen van <a>	98
Bladwijzers maken	101
Link naar een grote foto	102
Link naar het vervolg van een artikel	103
Link naar een e-mailadres	104
Link naar een telefoonnummer	104
Een pdf-bestand weergeven of downloaden	105
<b>Klik hier voor richtlijnen</b>	<b>105</b>
<b>Externe bestanden koppelen met &lt;link&gt;</b>	<b>108</b>
Stylesheet koppelen	108
Een favicon gebruiken	109
Opeenvolgende pagina's	110
Vooraf laden: prefetch, preload, prerender en dns-prefetch	111

<b>Knoppen en hyperlinks</b>	<b>113</b>
<b>Samenvatting</b>	<b>114</b>
<b>Oefeningen</b>	<b>114</b>
<b>5 Beeld, geluid en andere externe inhoud</b>	<b>115</b>
<b>Inleiding</b>	<b>116</b>
<b>URL's en structuur van de website</b>	<b>116</b>
Lokale site	118
<b>Soorten afbeeldingen</b>	<b>119</b>
Bitmaps	119
Vectorafbeeldingen: SVG	120
<b>Bestandsformaten voor bitmaps</b>	<b>122</b>
GIF	122
JPEG	123
PNG	124
WebP	125
<b>De juiste afbeelding voor het scherm</b>	<b>125</b>
<b>Afbeeldingen plaatsen met &lt;img&gt;</b>	<b>126</b>
De bron: src	127
Beschrijvende tekst: alt	127
Afmetingen: width en height	128
Klikbare gebieden: usemap	129
<b>Responsive images</b>	<b>130</b>
Srcset voor schermen met hoge resolutie	130
Srcset voor variabele afmetingen	131
Bijgesneden afbeeldingen: <picture>	133
Het type afbeelding selecteren	134
Het verschil tussen <img> en <picture>	134
<b>Externe HTML-inhoud: &lt;iframe&gt;</b>	<b>135</b>
Attributen van <iframe>	135
Beveiliging	137
<b>Insluiten met &lt;embed&gt; of &lt;object&gt;</b>	<b>138</b>
<b>Video in een webpagina</b>	<b>138</b>
Bestandstypen voor video	139
Een codec kiezen	139
Video coderen	140
<b>Het element &lt;video&gt;</b>	<b>140</b>
De bron: <source>	141
Ondertiteling met <track>	142
Titelbestanden: WebVTT	144
<b>Audio op de website gebruiken</b>	<b>145</b>
<b>Samenvatting</b>	<b>146</b>
<b>Oefeningen</b>	<b>147</b>

<b>6</b>	<b>Formulieren maken</b>	<b>149</b>
	<b>De rol van formulieren</b>	<b>150</b>
	Formulieren aan de clientkant	150
	De rol van de webserver	151
	<b>Een compleet formulier</b>	<b>152</b>
	<b>De basis van een formulier: &lt;form&gt;</b>	<b>156</b>
	Buiten <form> mag ook	156
	De verwerking: action	156
	Verzendwijze: method	156
	Codering: enctype	157
	Automatisch aanvullen: autocomplete	157
	Niet valideren: novalidate	158
	Direct invoeren: autofocus	159
	De naam: name	159
	Het doel: target	159
	Buitenspelelementen: form	159
	<b>Labels bij invoervelden</b>	<b>161</b>
	<b>De tabvolgorde</b>	<b>162</b>
	<b>Het element &lt;input&gt;</b>	<b>162</b>
	Attributen van <input>	163
	<b>Soorten invoer: het attribuut type</b>	<b>170</b>
	Telefoonnummers: type="tel"	170
	E-mailadressen: type="email"	171
	Webadressen: type="url"	171
	Wachtwoorden: type="password"	171
	Getallen: type="number"	172
	Een bereik: type="range"	172
	Kleuren: type="color"	173
	Datum en tijd	174
	Selectievakjes: type="checkbox"	175
	Keuzerondjes: type="radio"	176
	Bestanden uploaden: <type="file">	178
	Verzendknop: type="submit"	178
	Resetknop: type="reset"	179
	Afbeeldingknop: type="image"	179
	Functieloze knop: type="button"	180
	Verborgene waarden: type="hidden"	180
	<b>Het element &lt;button&gt;</b>	<b>180</b>
	<b>Kiezen uit een lijst: &lt;select&gt;, &lt;datalist&gt;, &lt;option&gt;</b>	<b>181</b>
	<option>	182
	<select>	183
	<datalist>	184
	<b>Uitgebreide tekstinvoer: &lt;textarea&gt;</b>	<b>186</b>

	Groeperen met <code>&lt;fieldset&gt;</code> en <code>&lt;legend&gt;</code>	186
	Voortgang tonen: <code>&lt;progress&gt;</code>	189
	Schaal: <code>&lt;meter&gt;</code>	190
	Resultaat: <code>&lt;output&gt;</code>	191
	Samenvatting	192
	Oefeningen	193
<b>7</b>	<b>Tabellen maken</b>	<b>195</b>
	Inleiding	196
	De structuur van een HTML-tabel	196
	De basis: <code>&lt;table&gt;</code>	198
	Rijen en kolommen: <code>&lt;tr&gt;</code> en <code>&lt;td&gt;</code>	198
	Koptekst: <code>&lt;thead&gt;</code>	199
	Een bijschrift: <code>&lt;caption&gt;</code>	200
	Overige elementen	201
	De tabelinhoud: <code>&lt;tbody&gt;</code>	201
	Voettekst: <code>&lt;tfoot&gt;</code>	201
	Kolomgroepen: <code>&lt;colgroup&gt;</code> en <code>&lt;col&gt;</code>	202
	Voorbeelden van tabellen	203
	Samenvatting	206
	Oefeningen	207
<b>8</b>	<b>De basis van CSS</b>	<b>209</b>
	Wat CSS is	210
	Voorgeschiedenis	210
	En toen was er CSS3	211
	CSS3 is gewoon CSS	211
	Waarom CSS zo handig is	213
	De taal CSS	213
	Waarden en eenheden	215
	Globale waarden	217
	Lengte	217
	De pixel	217
	De eenheid em	218
	De makkelijke em: rem	218
	Handig voor tekstvakken: ch	219
	Eenheden gebaseerd op de viewport	219
	Overige eenheden	220
	Samenvatting afmetingen	220
	CSS-verwerking door de browser	221
	De boomstructuur	221
	Waarden toewijzen	222

De cascade	223
Specificiteit berekenen	224
Overerving	226
Percentages werken door	227
<b>De plaats van de CSS-declaraties</b>	<b>228</b>
Het attribuut style	229
Het element <style>	229
Extern bestand: <link>	230
<b>Het CSS-bestand</b>	<b>231</b>
Commentaar toevoegen	231
Indeling van CSS-bestanden	232
Andere stijlbladen importen	232
<b>De ingebouwde stijlen aanpassen</b>	<b>232</b>
<b>Elementen benaderen met selectors</b>	<b>234</b>
<b>Elementnaam, klasse of beide?</b>	<b>235</b>
Elementnamen en selectors	235
Klassen	237
Van alles wat	238
En de winnaar is...	239
<b>Basisselectors</b>	<b>239</b>
Typeselector	240
Universele selector *	240
Klasseselector	242
ID-selector	243
Attribuutselectors	243
Pseudoklassen	244
<b>Pseudo-elementen</b>	<b>248</b>
Pseudo-elementen level 4	250
<b>Combinatieselectors: afstamming, kind en sibling</b>	<b>251</b>
Geen combinator: komma	251
Algemene afstamming: spatie	251
Kindselector	252
Aangrenzend: +	252
Op hetzelfde niveau: ~	252
<b>Best practices</b>	<b>253</b>
<b>Samenvatting</b>	<b>254</b>
<b>Oefeningen</b>	<b>255</b>
<b>9 Lay-out: boxmodel en weergavemodel</b>	<b>257</b>
<b>De opbouw van pagina's</b>	<b>258</b>
Browserstijlen zijn er niet voor niets	258
<b>Het boxmodel</b>	<b>259</b>
De eigenschap box-sizing	260

## Inhoud

Waarden voor breedte en hoogte	263
Begrensdte breedte en hoogte	264
<b>Marges</b>	<b>265</b>
Horizontaal centreren	265
Verticaal centreren	266
Negatieve marge	267
Samengevoegde marges	267
<b>Padding</b>	<b>268</b>
<b>Randen</b>	<b>269</b>
Randdikte	269
Randkleur	269
Transparante rand	270
Randstijl	271
Korte notatie voor randen	272
<b>Overlopende inhoud: overflow</b>	<b>273</b>
<b>Overloop van de scrol</b>	<b>275</b>
<b>De toekomst: logische eigenschappen</b>	<b>276</b>
Breedte en hoogte	277
Het einde van top, bottom, left, right?	277
Wat is het nut?	278
Praktische beperking	279
<b>De toekomst: box alignment</b>	<b>279</b>
<b>Weergavemodel</b>	<b>280</b>
Achter de schermen	281
<b>De eigenschap display</b>	<b>281</b>
Waarden voor binnen en buiten	281
<b>Inhoud (niet) weergeven: visibility</b>	<b>285</b>
Niet verbergen voor screenreaders	285
<b>Samenvatting</b>	<b>286</b>
<b>Oefeningen</b>	<b>287</b>
<b>10 Lay-outs maken met CSS</b>	<b>289</b>
<b>Inleiding</b>	<b>290</b>
<b>Responsive design</b>	<b>290</b>
Media queries	291
Schermresolutie	294
Portret of landschap	294
Beeldverhouding	295
Weergave op mobiele schermen: de metatag viewport	295
CSS-pixels en hardwarepixels (device pixels)	297
<b>Lay-out: positionering</b>	<b>297</b>
Het omvattende blok	297
Positioneringsschema's	298



De eigenschap position	298
Verschuiven met top, right, bottom en left	299
Verschuiven met logische waarden	299
Position: static	299
Position: relative	299
Position: absolute	300
Position: fixed	302
Position: sticky	303
Float en clear	305
Float als hulpmiddel voor lay-outs	307
Block formatting context	310
<b>Lay-out: multi-column</b>	<b>312</b>
Kolommen instellen	312
Marges worden niet samengevoegd	314
Kolommen overspannen	315
Ruimte tussen de kolommen	316
Lijn tussen de kolommen	317
Kolommen vullen	318
Afbreken in kolommen	318
Wanneer multi-column (niet) nuttig is	321
Toepassing: bol.com	322
Toepassing: portfolio	324
<b>Lay-out in een raster: gridlay-out</b>	<b>326</b>
Kenmerken van gridlay-out	327
Hulpmiddelen voor gridlay-out en flexbox	328
Een raster maken	330
De functie repeat()	331
De eenheid fr	331
Rasteritems plaatsen op lijnummers	332
Kolommen en rijen samen in grid-area	335
De lijnen een naam geven	336
Items plaatsen in gebieden met een naam	338
Afmetingen van tracks instellen	340
Ruimte tussen tracks	341
Het raster zichzelf laten bouwen	342
Het verschil tussen auto-fill en auto-fit	345
Van richting veranderen met grid-auto-flow	345
De gaten in het raster opvullen	346
Uitlijnen in gridlay-out	348
Toepassing: nos.nl	350
<b>Lay-out: flexbox</b>	<b>359</b>
Kenmerken van flexbox	359
Het assenstelsel van flexbox	359
Flex-direction	360

## Inhoud

Flex-wrap	361
Flex-flow	362
Flexcontainers en flexitems	362
Items flexibel maken met de eigenschap flex	363
Instellingen voor flex	364
Samenvatting: het formaat van flexitems	365
Items uitlijnen met automarge	366
Uitlijnen op de hoofdas: justify-content	367
Uitlijnen op de kruisas: align-self	369
Regels uitlijnen: align-content	370
Toepassing: navigatiebalk	371
Toepassing: kaarten	374
Toepassing: nos.nl	378
<b>Samenvatting</b>	<b>380</b>
<b>Oefeningen</b>	<b>381</b>
<b>11 Tekst en typografie</b>	<b>385</b>
<b>Inleiding</b>	<b>386</b>
<b>Lettertype: van systeem of online?</b>	<b>386</b>
Web fonts	387
<b>Lettertypen downloaden: @font-face</b>	<b>389</b>
Eigenschappen van @font-face	390
Een embedcode gebruiken	393
<b>Eigenschappen voor lettertype: font</b>	<b>395</b>
Lettertype	395
Lettergrootte: waarden en overwegingen	396
Geavanceerd: responsieve tekst	397
Toepassing van font-size	398
Grootte aanpassen: font-size-adjust	399
Regelhoogte: line-height	399
Letterstijl: font-style	401
Lettergewicht: font-weight	401
De verzameleigenschap font	402
Uitrekken of indrukken: font-stretch	403
Kleinkapitaal: font-variant	404
<b>Eigenschappen voor tekst</b>	<b>405</b>
Uitlijnen: text-align en text-align-last	405
Lijneffecten op tekst	405
Inspringen	407
Rechtopstaande tekst: text-orientation	408
Overlopende tekst: text-overflow	409
Tekst met schaduw	409
Tekst omzetten: text-transform	411

Witruimte behouden: white-space	411
Ruimte tussen woorden en letters	412
Afbreken: word-break en overflow-wrap	413
<b>Opmaak van lijsten</b>	<b>414</b>
Opsommingstekens	415
<b>Lijst als navigatie</b>	<b>416</b>
<b>Samenvatting</b>	<b>422</b>
<b>Oefeningen</b>	<b>423</b>
<b>12 Kleur, randen en achtergronden</b>	<b>425</b>
<b>Inleiding</b>	<b>426</b>
<b>Kleurwaarden</b>	<b>426</b>
RGB en RGBA	427
HSL en HSLA	428
Transparantie	430
<b>Kleur van tekst</b>	<b>430</b>
<b>De achtergrond</b>	<b>431</b>
Achtergrondkleur	431
Achtergrondaafbeelding	431
background-image	432
background-repeat	432
background-attachment	433
background-position	433
background-clip	435
background-origin	436
background-size	437
background	438
Verschillende achtergrondaafbeeldingen	439
Kleurverloop als achtergrond	440
<b>Kleurverlopen</b>	<b>441</b>
Lineair verloop	441
Radiaal verloop	443
Repeterende verlopen	444
<b>Afgeronde hoeken</b>	<b>445</b>
<b>Randafbeelding</b>	<b>447</b>
border-image-source	448
border-image-slice	448
border-image-width	448
border-image-outset	449
border-image-repeat	449
Voorbeeld van een randafbeelding	449
<b>Schaduw</b>	<b>450</b>
De werking van schaduw	451

Material design	452
<b>Afbeeldingen uitsnijden met clip-path</b>	<b>453</b>
Toepassing: header met schuine kanten	456
<b>Beeldfilters</b>	<b>458</b>
<b>Kleuren mengen</b>	<b>458</b>
Isolation	460
<b>Samenvatting</b>	<b>460</b>
<b>Oefeningen</b>	<b>461</b>
<b>13 Overgangen en animaties, calc(), attr() en CSS-variabelen</b>	<b>463</b>
<b>Beweging in webpagina's</b>	<b>464</b>
<b>Overgangen (transitions)</b>	<b>464</b>
Kant-en-klare overgangen	467
Positie aanpassen	467
<b>Animatie</b>	<b>469</b>
Opbouw een animatie	469
Keyframes	471
<b>Transformatie</b>	<b>473</b>
Tweedimensionale transformaties	474
Schalend menu	476
Driedimensionale transformaties	478
Perspectief	479
Kijkrichting	480
3D-ruimte	480
<b>Rekenen met calc()</b>	<b>484</b>
Voorbeelden	485
<b>CSS-variabelen: de functie var()</b>	<b>485</b>
Kenmerken van variabelen	486
Variabelen declareren	487
Toepassingen van variabelen	487
<b>Attributen uitlezen met attr()</b>	<b>489</b>
<b>All reset alles, behalve variabelen</b>	<b>491</b>
<b>Samenvatting</b>	<b>491</b>
<b>Oefeningen</b>	<b>492</b>

# Webtalen, browsers en editors

**H**TML is de belangrijkste taal op internet. Sinds 2011 werken we met HTML 5 en daar wordt nog steeds aan gesleuteld. Dit hoofdstuk beschrijft het ontstaan van HTML 5 en de actuele situatie. Er wordt kort uitgelegd wat HTML en CSS zijn en wat HTML 5 is. Er is een overzicht van browsers en hulpmiddelen voor het schrijven van HTML-documenten en u maakt kennis met handige gereedschappen.

U leert in dit hoofdstuk:

*Hoe we hier gekomen zijn.*

*De rol van HTML en van CSS, en wat HTML 5 is.*

*Welke browsers niet mogen ontbreken in uw gereedschapskist.*

*Handige gereedschappen: editors, validators en hulpmiddelen voor ontwikkelaars.*

## Introductie

Het world wide web met zijn blogs, games, webwinkels, videosites, sociale media, magazines en al die andere soorten sites en apps zou niet bestaan zonder de taal HTML, de afkorting van Hypertext Markup Language. Het fraaie uiterlijk wordt gemaakt met CSS (Cascading Style Sheets) en de logica wordt geprogrammeerd met JavaScript, maar zonder HTML was er niets.

Dit boek gaat over HTML 5. Bij de publicatie van deze editie is de laatste voltooide versie HTML 5.2 uit december 2017 en wordt er gewerkt aan HTML 5.3.

HTML 5 kan niet los worden gezien van de voorgangers, HTML 4.01 en XHTML. HTML 5 is daar een voortzetting van, waarbij onderdelen zijn aangepast, toegevoegd en afgeschaft, maar alle oude webpagina's blijven werken. Afschaffen betekent in dit verband dan ook dat auteurs worden aangemoedigd nieuwe webpagina's volgens nieuwe standaarden te coderen en oude pagina's te moderniseren. Doen ze dat niet, dan blijven de webpagina's toch toegankelijk voor bezoekers. Ze zien die er misschien niet meer zo uit als in 2000 werd bedoeld, maar de browser moet in elk geval de inhoud weergeven.

## Een korte terugblik

Toen de eerste editie van dit boek verscheen – voorjaar 2011 – was de inkt van het logo van HTML 5 nog nat. De specificatie was nog volop in ontwikkeling, maar de belangrijkste elementen waren al omarmd door de gemeenschap van webbouwers en door browserfabrikanten. In 2014 heeft de beheerder van de webtalen, het World Wide Web Consortium (W3C) de specificatie 'definitief' vastgesteld (in W3C-termen is het dan de aanbevolen specificatie, gelabeld als *recommendation*). Definitief staat niet voor niets tussen aanhalingstekens, want het werk aan de specificatie gaat gewoon door. Het betekent alleen dat die versie wordt bevroren en dat het werk verder gaat in de volgende versie. In november 2016 werd HTML 5.1 de aanbevolen versie en in oktober 2017 werd dat HTML 5.1 2nd edition. Snel daarna, in december 2017, werd HTML 5.2 de aanbevolen versie en is het werk aan HTML 5.3 begonnen. Bent u er nog?

Elke revisie bevat kleine veranderingen. Vaak zijn het aanvullingen en verbeteringen gebaseerd op hoe HTML wordt gebruikt en op reacties uit de gebruikersgemeenschap. Soms vervalt een element (omdat het nauwelijks wordt gebruikt) of verandert de toepassing ervan. *Breaking changes* zijn in elk geval niet verwachten, omdat het web nu eenmaal niet mag 'breken'.



**Afbeelding 1.1** *Het logo van HTML 5, in 2011 gepresenteerd door het W3C.*

Naast de versie van het W3C is er een tweede HTML-specificatie en die gaat door het leven als *HTML Living Standard*. Een updatedatum geeft aan wanneer die voor het laatst is bijgewerkt. Hierbij zijn er geen bevroren versies, het is een ‘levende’ standaard waar continu aan wordt gewerkt.

# HTML

## Living Standard — Last Updated 14 March 2019

**Afbeelding 1.2** *HTML Living Standard, beheerd door WHATWG.*

Dat er twee standaarden bestaan klinkt misschien vreemd (en dat is het ook), maar het komt doordat de ontwikkeling van HTML een tijdje heeft stilgelegen.

### Afspraken over webtalen

Sinds 1994 wordt geprobeerd lijn te brengen in de talen waarmee websites (en andere webtoepassingen) worden gemaakt. Die talen zijn onder meer Hypertext Markup Language (HTML) en Cascading Style Sheets (CSS). Een standaard voor deze talen is geen overbodige luxe. Toen de belangstelling voor internet groeide en er meer browsers op de markt werden gebracht, deed elke fabrikant wat hem leuk leek. De HTML-standaard werd wel ingebouwd, maar daar werd een eigen draai aangegeven. Zo ontstond een situatie waarin browsers naast standaard-HTML ook eigen tags hadden (berucht waren `<blink>` voor knipperende tekst en `<bgsound>` voor achtergrondgeluid). Bij CSS was de situatie vergelijkbaar. Zo kon een website met dezelfde code er in elke browser anders uitzien, anders werken of compleet uit elkaar vallen. Vooral Microsoft heeft met eigen interpretaties van de standaarden in Internet Explorer ontwikkelaars tot wanhoop gedreven. Intussen zijn dergelijke problemen grotendeels opgelost en werken de talen in alle browsers vrijwel hetzelfde.

Voor het beheer van de talen is het World Wide Web Consortium opgericht, kortweg W3C. In verschillende werkgroepen wordt overlegd door browserfabrikanten zoals Google, Microsoft, Mozilla en Opera. Omdat W3C ook allerlei andere webstandaarden beheert, vindt u onder de leden ook bedrijven als Bloomberg, Vrije Universiteit Amsterdam, Nokia en Walt Disney Company. Het doel van bijvoorbeeld de werkgroep Web Platform is afspraken maken over hoe een auteur in een webdocument aangeeft dat iets een hyperlink, een kop of een opsomming is, en hoe een browser daarmee zou moeten omgaan.



### World Wide Web Consortium

Kijk voor meer informatie over wat het W3C is en doet op [w3.org](http://w3.org). Dat is de thuishaven van deze in Amerika gevestigde organisatie. U vindt er informatie over elke webstandaard. Daar hoort JavaScript niet bij, dat wordt beheerd door Ecma ([www.ecma-international.org](http://www.ecma-international.org)).

The screenshot shows the W3C website's 'ALL STANDARDS AND DRAFTS' page. The page has a dark header with the W3C logo and navigation links for STANDARDS, PARTICIPATE, MEMBERSHIP, and ABOUT W3C. Below the header, there's a sidebar with 'FEATURED SPECIFICATIONS' including Service workers, CSS Grid, ARIA in HTML, and CSP. The main content area displays a list of 1170 specs with search filters for Title, Tag, Status, and Version. A 'Reset' button is visible. The list includes items like 'The Screen Orientation API', 'Trace Context Protocols Registry', 'Payment Method: Basic Card', 'User Timing Level 3', 'Navigation Timing Level 2', 'Generic Sensor API', 'Server Timing', 'Ambient Light Sensor', and 'Gyroscope'. Each item shows its working group, draft status, date, and authors.

**Afbeelding 1.3** Een (heel) klein deel van de standaarden en concepten op de website van W3C.

Het is allerm minst vanzelfsprekend dat er nu een W3C-standaard van HTML 5 is. Eind jaren negentig, de standaard HTML 4.01 was klaar, besloot het W3C zich volledig te richten op XHTML, een vorm van HTML gemodelleerd naar de strikte regels van XML. Het begon met het herschrijven van HTML naar XML (XHTML 1.0) en tegelijk werd gewerkt aan een geheel nieuwe versie (XHTML 2)



die niet uitwisselbaar zou zijn met eerdere HTML- en XHTML-talen. Er was daarmee een route gekozen waarbij het bestaande web zou blijven zoals het was (HTML 4) en er daarnaast een niet-uitwisselbaar nieuw web zou ontstaan met op XML gebaseerde applicaties.

Tijdens dat ontwikkelproces zagen Mozilla en Opera in 2004 nieuwe mogelijkheden voor HTML en zij hebben voorgesteld om toch door te gaan met de ontwikkeling van HTML, maar daar waren bij het W3C niet genoeg medestanders voor. In reactie daarop besloten Apple, Mozilla en Opera dan maar zelf aan de slag te gaan met HTML 5. Daarvoor richtten zij in 2004 de Web Hypertext Application Technology Working Group op, kortweg WHATWG. Hun voornaamste uitgangspunt was dat webontwikkelaars een webtaal gebaseerd op het oude HTML 4 tot hun beschikking zouden krijgen, dus uitwisselbaar, maar met nieuwe mogelijkheden om een volwaardig alternatief te kunnen bieden voor *native* webapps (iOS en Android) en gepatenteerde mediatechnieken zoals Flash van Adobe en Silverlight van Microsoft. Dit initiatief viel bij de gemeenschap van webontwikkelaars in goede aarde. Ontwikkelaars ontvingen de ontwikkelingen met gejuich en browserfabrikanten gingen de nieuwe ideeën inbouwen in hun browsers, en daardoor ging uiteindelijk ook bij het W3C het roer om. De werkgroep XHTML is eind 2009 opgeheven en ook bij het W3C gaat alle aandacht weer uit naar HTML.

Het zijn de gedrevenheid van WHATWG en de snelheid waarmee in browsers nieuwe mogelijkheden worden ingebouwd die de ontwikkeling van HTML in een stroomversnelling hebben gebracht. Google (Chrome), Mozilla (Firefox), Opera en Microsoft (Edge) brengen in hoog tempo updates voor de browser uit. Het resultaat is dat de nieuwste versies van alle toonaangevende browsers veel nieuwe mogelijkheden ondersteunen. Het werkt als een vliegwiel: het feit dat nieuwe opties snel worden opgenomen in browsers stimuleert de ontwikkeling.



#### Automatische updates

Er worden niet alleen sneller nieuwe versies van browsers uitgebracht, ze worden ook veel sneller op grote schaal gebruikt door het internetpubliek. Dat komt doordat browserupdates automatisch worden gedownload en geïnstalleerd, zonder tussenkomst van de gebruiker.

---

## Twee smaken HTML

Zijn er nu twee smaken HTML? In zekere zin is dat zo. Hoewel de oorspronkelijke HTML 5-aanbeveling op de website van het W3C het resultaat is van samenwerking tussen W3C en WHATWG, hebben de wegen van de groepen zich na 'HTML 5.0' gescheiden. Eerst leek het erop dat W3C HTML 5 helemaal

niet verder zou ontwikkelen, maar intussen weten we beter. WHATWG gaat ook gewoon door en zo lopen er twee HTML-specificaties naast elkaar. Wat is de zin daarvan en vooral: wat moet de webontwikkelaar ermee?

Hoewel W3C en WHATWG van mening verschillen over het omgaan met (web)standaarden, zijn er niet twee totaal verschillende HTML-standaarden. Een webontwikkelaar merkt er nauwelijks iets van. Zijn enige zorg is of een bepaalde optie goed werkt in de browser en het is niet moeilijk om daar achter te komen. (U mag alle websites vergeten, maar onthoud altijd deze twee: [caniuse.com](http://caniuse.com) en [developer.mozilla.org](http://developer.mozilla.org).)

Uiteindelijk profiteren webontwikkelaars en internetgebruikers alleen maar van de situatie. De clubs houden elkaar en de browserfabrikanten scherp, zodat een constante stroom van verbeteringen en nieuwe mogelijkheden in stand wordt gehouden. Dat houdt het web levend en vitaal, en het leven van een webontwikkelaar boeiend.



### Bekijk de specificaties

Uiteraard zijn alle specificaties online beschikbaar. De actuele W3C-aanbeveling is te vinden op [www.w3.org/TR/html](http://www.w3.org/TR/html). Werk in uitvoering (het concept van de volgende revisie) is te vinden via de pagina [www.w3.org/TR/](http://www.w3.org/TR/) met als filters de tag HTML en de status Working Draft. De specificatie van WHATWG is te vinden op [html.spec.whatwg.org/multipage/](http://html.spec.whatwg.org/multipage/) en toont altijd de versie van de dag.

W3C Working Draft

## HTML 5.3

W3C Working Draft, 18 October 2018



**This version:**  
<https://www.w3.org/TR/2018/WD-html53-20181018/>

**Latest published version:**  
<https://www.w3.org/TR/html53/>

**Editor's Draft:**  
<https://w3c.github.io/html/>

**Afbeelding 1.4** *Hier gaan we het uitgebreid over hebben.*

## Wat HTML 5 is

HTML 5 is eigenlijk niet meer dan een upgrade van HTML 4. We hebben het dan alleen over de markeertaal HTML. Maar HTML 5 wordt ook gebruikt als verzamelnaam voor webtechnieken om moderne (mobiele) webapplicaties te bouwen. Naast HTML gaat het dan ook over een verzameling definities waarmee de ontwikkelaar toegang krijgt tot onderdelen van de browser of

smartphone (API's, zie de opmerking hierna) en aanvullende technieken zoals CSS, JavaScript, XML, JSON en SVG. Maar deze horen niet bij de HTML 5-standaard zelf. Ze hebben wel een sterke relatie en het is bijna niet mogelijk om moderne websites en webapps te bouwen zonder er een of meer te gebruiken. Op [www.w3.org/TR/](http://www.w3.org/TR/) kunt u zoeken met de tag Web API om een overzicht te bekijken. Ook op [spec.whatwg.org](http://spec.whatwg.org) vindt u een overzicht van technologieën voor het webplatform.

Over HTML en CSS leest u alles in dit boek. Voor de andere technieken moet u andere bronnen raadplegen. Een korte toelichting op de functie ervan:

- JavaScript is een programmeertaal (scripttaal) om onder meer webpagina's interactief te maken: (onder)delen van de pagina updaten, gegevens lezen uit een databron (JSON), een winkelwagen bijhouden, cookies plaatsen en lezen enzovoort.
- XML staat voor Extensible Markup Language en is een markeringstaal (zoals HTML). XML is ontwikkeld om gestructureerde gegevens uit te wisselen tussen (web)toepassingen op een manier die ook voor mensen leesbaar is.
- JSON is de afkorting van JavaScript Object Notation en is een veelgebruikt alternatief voor XML. De functie is hetzelfde: uitwisseling van gegevens tussen een databron en een applicatie. JSON wordt verder niet besproken.
- SVG staat voor Scalable Vector Graphics. Dit is een op XML gebaseerd bestandsformaat voor vectorafbeeldingen (vectorafbeeldingen kunnen worden vergroot of verkleind zonder dat ze rafelig of blokkerig worden zoals bitmapafbeeldingen). Ook SVG valt buiten het bestek van dit boek.



### Toegangspunt voor ontwikkelaars: API

De afkorting API staat voor application programming interface. Het is een toegangspunt voor aanvullende functies waarbij de ontwikkelaar eenvoudig toegang heeft tot die functies zonder dat hij precies hoeft te weten ze werken. Hij hoeft alleen te weten hoe hij toegang krijgt. Een vergelijking kan zijn het bedienen van software met een grafische interface of met getypte opdrachten in een terminalvenster. Voorbeelden zijn API's voor het opnemen van beeld en geluid met een telefoon, voor het werken met een muis, pen of aanraakscherm, voor het werken met de geografische locatie van de gebruiker, of voor het opslaan van gegevens op het apparaat van de gebruiker. Hierbij is altijd JavaScript nodig.

---

## Kennismaken met HTML en CSS

Lees zeker de volgende paragrafen als HTML en het maken van webpagina's voor u volledig onbekend terrein zijn. Hebt u al wat ervaring met HTML en CSS, dan kunt u ze eventueel overslaan.

Het eerste deel van dit boek gaat over HTML. Maar tenzij u kale tekstpagina's wilt maken, kunt u niet zonder CSS, wat staat voor Cascading Style Sheets. Deze taal is ontwikkeld om aan te geven hoe de met HTML gemarkeerde teksten (en afbeeldingen en meer) eruit moeten zien en hoe de webpagina moet worden ingedeeld, de lay-out. Daarom gaat het tweede deel van het boek over CSS.

Naast HTML en CSS kunnen programmeertalen worden gebruikt om webpagina's tot leven te brengen. JavaScript is zo'n programmeertaal. Deze wordt bijvoorbeeld gebruikt om acties na een klik op een knop uit te voeren of om onderdelen van een webpagina te manipuleren. Maar ook complete apps worden gebouwd met JavaScript-frameworks zoals Angular, React of Vue.js. Het is belangrijk om u ook in JavaScript te verdiepen als u verder wilt met webontwikkeling.

HTML, CSS en ook JavaScript doen hun werk in de browser op het apparaat waarmee een bezoeker uw pagina's bekijkt. Er is natuurlijk communicatie met de webserver, maar het werk wordt gedaan door de browser. Daarmee vormen uw bezoeker, zijn computer/tablet/telefoon en vooral zijn browser de client-side.

### Aan de serverkant

Er zijn ook talen en technieken waarbij het werk op de webserver wordt gedaan, server-side dus, bijvoorbeeld het bij elkaar zoeken van componenten waaruit een webpagina wordt samengesteld. De server stuurt dan een kant-en-klare webpagina naar de client (de browser). Een veelgebruikte programmeertaal op webserver is PHP. Deze taal wordt bijvoorbeeld toegepast om een contentmanagementsysteem (cms) zoals WordPress ([www.wordpress.org](http://www.wordpress.org)) of Joomla! ([www.joomla.org](http://www.joomla.org)) aan te sturen. PHP wordt daarbij onder meer gebruikt om pagina's samen te stellen aan de hand van informatie die in een database is opgeslagen. PHP blijft in dit boek verder buiten beschouwing.



#### Nog meer talen

Er zijn nog veel meer talen voor het web dan de hier genoemde, maar tenzij u specialistische toepassingen wilt ontwerpen, krijgt u daar niet mee te maken. Op [w3c.org](http://w3c.org) kunt u er echter van alles over vinden.

---



## JavaScript op de server

Met de opkomst van JavaScript-frameworks zoals Angular, React en Vue.js is ook de rol van JavaScript als servertaal belangrijker geworden. Daarbij spelen het platform Node.js ([www.nodejs.org](http://www.nodejs.org)) en de pakketbeheerder npm ([www.npmjs.com](http://www.npmjs.com)) een centrale rol. Met Node.js kunnen toepassingen worden geschreven in JavaScript (en andere talen die naar JavaScript kunnen worden omgezet, zoals TypeScript). Node.js bevat onder meer een webserverfunctie, waarmee op de server webpagina's en apps kunnen worden samengesteld uit componenten die zijn gebouwd met de genoemde frameworks (*server side rendering*). Dit in tegenstelling tot pure HTML-pagina's die u kant en klaar op de server zet. In dit boek kunnen al die mogelijkheden niet worden uitgelegd, maar u weet nu dat er na HTML en CSS nog veel meer te ontdekken is.

## De functie van HTML

HTML staat voor Hypertext Markup Language. Met HTML worden de *structuur* van de pagina en de *betekenis* van de elementen in die pagina aangegeven. (In Engelstalige bronnen wordt gesproken over *structure* en *semantics*). Er wordt met tags (labels) beschreven wat de kopteksten en alineateksten van een pagina zijn en welke afbeeldingen in de pagina moeten worden geladen, en met hyperlinks wordt aangegeven wat de onderlinge relatie tussen pagina's is. HTML heeft niets te maken met het uiterlijk van een pagina.

HTML is geen programmeertaal. Het is een markeertaal (*markup*). Vandaar ook de afkorting Hypertext Markup Language, oftewel 'een markeertaal voor hypertext'. Hypertekst is tekst met hyperlinks (koppelingen) naar andere tekst, afbeeldingen, audio, video en meer.

De hyperlinks, maar ook alle andere onderdelen van een webpagina, zijn de *elementen* van HTML en die worden gemarkeerd met tags. Om met HTML te kunnen werken, moet u weten welke elementen er zijn en welke tags daar bij horen. Stel dat u een artikel hebt met een kop en een alinea. De HTML-code hiervoor kan zijn:

```
<article>
  <h1>De functie van HTML</h1>
  <p>HTML staat voor Hypertext Markup Language. Met HTML worden de <i>structuur</i> van de pagina
  en de <i>betekenis</i> van de elementen in die pagina aangegeven.</p>
</article>
```

Een artikel wordt op een HTML-pagina aangeduid met het element `article`. Het begin van het artikel wordt gemarkeerd met de openingstag `<article>`.

Voor de belangrijkste kop op de pagina heeft HTML het element `h1`. Het begin van de koptekst is gemarkeerd met de openingstag `<h1>`. Alles wat hierna volgt (in dit voorbeeld de tekst *De functie van HTML*) wordt gemarkeerd als koptekst (met de `h` van *header*). De kop loopt door tot de sluittag `</h1>`. Een stuk gemarkeerde tekst wordt altijd afgesloten met een sluittag. Een sluittag heeft dezelfde naam als de openingstag en begint met een voorwaartse slash `</...>`.

Hetzelfde geldt voor de alineatekst. De openingstag `<p>` markeert het begin van de alinea (*paragraph*) en de sluittag `</p>` het einde ervan.

In de alinea ziet u nog meer tags, twee keer de openingstag `<i>` en de sluittag `</i>`. Deze markeren belangrijke begrippen. In dit boek zijn die woorden om dezelfde reden cursief. Toch staat `<i>` niet voor *italic* (cursief). 'Vroeger', toen HTML-tags ook opmaak markeerden, was dat wel zo, maar nu markeert het element `i` tekst die op de een of andere manier afwijkt van de rest, meer daarover in hoofdstuk 2. De tags `<i>` en `</i>` staan binnen de alineatags `<p>`. Dit insluiten of nesten van tags is een belangrijk principe in HTML. Overigens kunnen niet alle tags worden genest. Hoe dat zit komt in volgende hoofdstukken aan de orde.

Het artikel wordt afgesloten de sluittag `</article>`. Alles wat tussen de begin- en eindtag `<article>` staat, hoort bij het HTML-element `article`. Dat 'doet' verder niets, het geeft alleen aan dat alles wat tussen de tags staat bij het artikel hoort. Op die manier is ook duidelijk dat de elementen `h1` en `p` bij elkaar horen. In de browser ziet het voorbeeld er zo uit als in de afbeelding.

HTML bevat meer dan honderd elementen. In dit boek worden de meeste ervan besproken.

### De functie van HTML

HTML staat voor Hypertext Markup Language. Met HTML worden de *structuur* van de pagina en de *betekenis* van de elementen in die pagina aangegeven.

**Afbeelding 1.5** Een artikel met een kop en een alinea met benadrukte woorden. Het uiterlijk is bepaald door de browser. Deze bevat een basisopmaak voor elk HTML-element.

## De functie van CSS

Met CSS wordt het uiterlijk van de pagina bepaald. CSS-code beschrijft hoe de kopteksten, alineateksten en alle andere elementen op de pagina worden opgemaakt. In CSS wordt aangegeven welk lettertype wordt gebruikt, welke lettergrootte, letterkleur, uitlijning, regelafstand, witruimte tussen onderdelen

en veel meer. Met CSS wordt ook de lay-out van de pagina ingesteld: kolommen, header, footer, kaders enzovoort. Zelfs transformaties, overgangen en animaties zijn mogelijk. CSS heeft dus alles te maken met vormgeving, maar niets met de inhoud van de pagina.

Neem nog even het voorbeeld van het artikel uit de voorgaande paragraaf in gedachten. In HTML wordt aangegeven dat een bepaalde tekst een koptekst moet zijn, maar dat zegt nog niets over het uiterlijk. Welk lettertype en welke kleur moet de koptekst hebben en hoe groot moet die zijn? Dat is de taak van CSS. Onthoud het volgende:

- Met HTML wordt de *structuur* van de webpagina en de *betekenis* van de inhoud gemarkeerd (kopteksten, tussenkoppen, hyperlinks, alinea's, invulvelden, knoppen en dergelijke).
- Met CSS wordt het *uiterlijk* van de webpagina ingesteld (lettertype, kleur, regelafstand, marges, en dat alles desgewenst per onderdeel van de pagina).

Met een paar regels CSS wordt het voorbeeld uit afbeelding 1.5 opgemaakt:

```
h1 {
  color: hsl(0, 0%, 15%);
  font-family: 'Source Sans Pro', sans-serif;
  font-weight: 600;
}
p {
  color: hsl(0, 0%, 5%);
  font-family: 'Source Serif Pro', serif;
  font-weight: 300;
}
```

## De functie van HTML

HTML staat voor Hypertext Markup Language. Met HTML worden de *structuur* van de pagina en de *betekenis* van de elementen in die pagina aangegeven.

**Afbeelding 1.6** *Hetzelfde artikel als in vorige afbeelding, opgemaakt met CSS. In de HTML-code is niets gewijzigd, alleen het uiterlijk is aangepast met enkele stijlregels.*

Het lettertype, het gewicht (de dikte) van de tekst en tekstkleur zijn ingesteld. De tekstgrootte is niet aangepast, maar doordat een ander lettertype is gekozen, is de tekst toch wat groter. Dit zijn eenvoudige aanpassingen en met CSS is heel veel meer mogelijk. De complete CSS-familie bestaat uit enkele honderden eigenschappen die samen alle mogelijke manieren bieden om een webpagina op te maken. Hoe meer CSS-eigenschappen u kunt toepassen, hoe meer mogelijkheden u hebt voor een fraai eindresultaat.



### CSS3

Net als de structuurtaal HTML is de opmaaktaal CSS sterk in ontwikkeling. Er wordt al jaren gewerkt aan de nieuwste standaard: CSS3, hoewel die naam eigenlijk niet klopt. Een groot deel van de oorspronkelijke specificatie is verdeeld in modules en er komen ook nieuwe modules bij. Al die modules worden onafhankelijk van elkaar ontwikkeld en daardoor hebben ze ook verschillende versienummers (2, 3 maar ook 4). Vandaar dat 'CSS versie 3' niet bestaat; het W3C heeft het ook gewoon over CSS.

---

## Browsers

Om webpagina's te kunnen maken en de uitleg in dit boek te kunnen volgen, hebt u geen bijzondere programma's nodig. Het maakt ook niet uit of u werkt op een pc met Windows of Linux of op een Mac met macOS (al zullen de gebruikte programma's natuurlijk iets verschillen). Elk platform gebruikt dezelfde HTML- en CSS-code.

U hebt weinig nodig om aan de slag te kunnen gaan:

- een tekstverwerker (*editor*) om HTML- en CSS-bestanden te schrijven en op te slaan;
- een webbrowsier om te kunnen bekijken wat u maakt.

De browser is natuurlijk een belangrijk programma. Tijdens het coderen van webpagina's gebruikt u de browser om te bekijken hoe de eindgebruikers uw pagina's te zien krijgen. Gebruik bij voorkeur verschillende browsers, zodat u kunt controleren of uw pagina's ook in andere browsers goed werken. Bedenk daarbij dat in modern ontwerp verschillen in weergave tussen browsers acceptabel zijn zolang de bruikbaarheid en functionaliteit niet in het geding zijn. Verschillen ontstaan overigens zelden door de HTML-code (HTML gaat over betekenis, niet over uiterlijk), maar meestal door de CSS-code (CSS is voor de opmaak).



De meestgebruikte browsers op desktopcomputers zijn Chrome, Firefox en Edge/Internet Explorer. Op mobiel en tablets zijn dat Chrome en Safari. Hoewel het lastig is om exacte marktaandeelen te achterhalen, is de grote lijn duidelijk. Op desktop heeft Chrome een aandeel van bijna 55%, Firefox 12%, Internet Explorer 12%, Edge 8% en Safari 8%. Op mobiel/tablet is de verdeling anders: Safari 46%, Chrome 37% en Samsung Internet 10%. Het aandeel van de andere browsers is op mobiel verwaarloosbaar. Dit zijn cijfers voor Nederland van 2017 en 2018 (bron: [gs.statcounter.com](http://gs.statcounter.com)). Europees en wereldwijd gezien zijn er wel verschillen in aandeel, maar in grote lijnen komt de verhouding overeen.



### Verschillen tussen browsers

Het is nogal een open deur, maar uw website moet in elke browser goed werken. Dat wil zeggen dat alle tekst leesbaar moet zijn, links en formulieren goed werken enzovoort. Of het bedoelde ontwerp ook in elke browser tot op de pixel perfect moet zijn, hangt vooral af van uw opdrachtgever. Maar verschillen worden steeds vaker geaccepteerd, omdat er zo veel verschillende apparaten, browsers en schermafmetingen bestaan. Bovendien stellen de verschillen van vandaag weinig voor vergeleken bij de verschillen van 'vroeger', toen elke browserfabrikant zijn eigen draai aan HTML en CSS gaf.

## Google Chrome

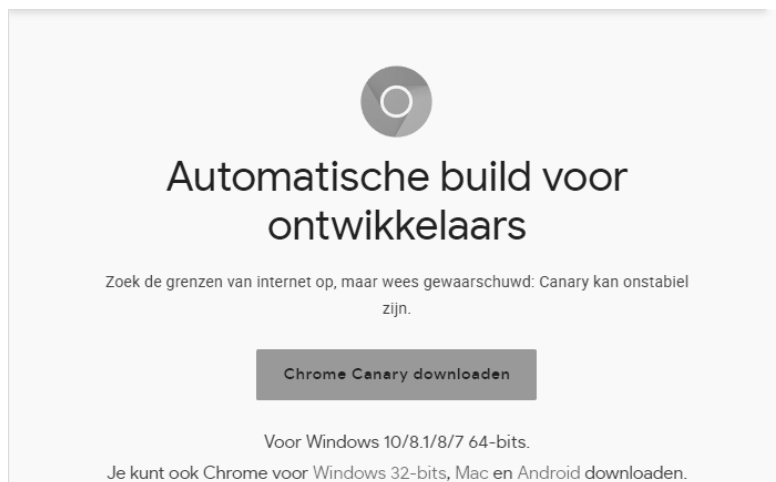
Het lijkt alsof hij er altijd al was, maar toch is Chrome, de browser van Google, pas eind 2008 gelanceerd. Tot die tijd was Internet Explorer de onbetwiste marktleider, al was het maar omdat de browser was vastgeklamd aan Windows. Chrome heeft intussen de strijd met Internet Explorer/Edge en Firefox beslist. Vanwege het grote marktaandeel mag deze browser dus niet op uw computer ontbreken. U kunt hem zelf downloaden en installeren vanaf [www.google.com/chrome](http://www.google.com/chrome). Chrome volgt de webstandaarden. De browsers werd eerst gebouwd op de engine WebKit, die ook in Safari van Apple wordt gebruikt. Sinds 2013 bevat Chrome een eigen versie van WebKit genaamd Blink.



### Browser engine

Een browser engine (ook wel *layout engine* of *rendering engine* genoemd) vertaalt de inhoud van een webpagina (de HTML) en informatie over opmaak (de CSS) naar een weergave op het scherm. De engine bepaalt dus welke onderdelen van HTML en CSS de browser kan gebruiken. Een CSS-eigenschap die niet in de programmacode van de engine zit, kan ook niet goed worden verwerkt. Er wordt dan gezegd dat de eigenschap niet wordt ondersteund. Bekende engines zijn WebKit (Safari), Blink (Chrome), Gecko (Firefox) en EdgeHTML (Edge).

Handig is de testversie van Chrome: Canary. Met deze ‘permanente bètaversie’ kunt u nieuwe functionaliteit testen. Canary wordt elke dag bijgewerkt en kan tegelijk met de stabiele versie van Chrome op uw computer worden uitgevoerd.



**Afbeelding 1.7** *Google Canary is aan de buitenkant niet anders dan Chrome, maar technisch loopt die versie altijd voor.*

Chrome kan eindeloos worden uitgebreid met extensies. Kijk op de site [chrome.google.com/extensions](https://chrome.google.com/extensions) waar u in de categorie Hulpprogramma's voor ontwikkelaars handige browseruitbreidingen vindt.

### Microsoft Edge (Internet Explorer)

Internet Explorer raakte na de komst van Chrome zijn traditionele status als meestgebruikte browser al kwijt en inmiddels is ook het levenseinde in zicht, want Internet Explorer is vervangen door de nieuwe browser van Microsoft: Edge. Internet Explorer had nooit een goede reputatie onder webontwikkelaars, omdat het van alle browsers de eigenzinnigste interpretatie van webstandaarden had. Ontwikkelaars hebben daardoor jarenlang de ene hack na de andere moeten bedenken om websites compatibel te laten zijn met de verschillende versies van deze browser. In versie 11 worden de standaarden overigens wel veel beter gevolgd (maar dat is dus een beetje laat).

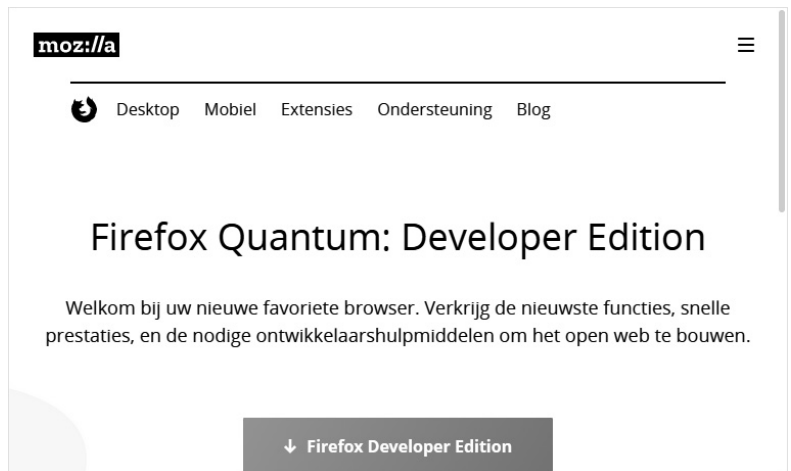
Bij Windows 10 is de nieuwe browser Edge geïntroduceerd. Deze biedt nagenoeg dezelfde goede ondersteuning van webstandaarden als de concurrenten. Het is een browser in ontwikkeling en er zijn vergeleken met Chrome en Firefox nog niet veel extensies/add-ons voor en dat aantal groeit niet snel. Regelmatig

verschijnt een update die de functionaliteit van Edge vergroot. De thuispagina van Microsoft Edge is te vinden op [www.microsoft.com/nl-nl/windows/microsoft-edge](http://www.microsoft.com/nl-nl/windows/microsoft-edge). De browser is niet als los programma beschikbaar.

Voor Edge heeft Microsoft een eigen browser engine ontwikkeld: EdgeHTML. Die heeft uiteraard geen last van het buggy verleden van Internet Explorer, maar moet wel van de grond af worden opgebouwd. Daardoor loopt Edge behoorlijk achter bij Chrome en Firefox, zeker bij de snelle vernieuwingen in CSS. Eind 2018 werd bekend dat Microsoft voor de desktopbrowser overstapt naar het open-sourceproject Chromium. Deze engine wordt onder meer in Opera gebruikt. Dit moet in de loop van 2019 merkbaar worden.

## Mozilla Firefox

Korte tijd leek wereldheerschappij binnen handbereik, maar ook Mozilla Firefox is overvleugeld door Chrome. Enkele jaren geleden nog leek Firefox de leidende positie van Internet Explorer te gaan overnemen. Het zijn nu nog steeds concurrenten, maar dan voor de lagere posities. Vanwege de uitstekende implementatie van webstandaarden was het de lieveling van ontwikkelaars. Firefox is ooit ontstaan uit de failliete boedel van de oerbrowser Netscape Navigator. Programmeurs die eerst voor Netscape werkten, hebben in eigen beheer en met samenwerking van de opensourcegemeenschap een nieuwe browser ontwikkeld. Deze browser werd Firefox genoemd en in het najaar van 2004 werd versie 1.0 uitgebracht. In 2006 verscheen Firefox 2.0. Binnen enkele weken na de introductie was het programma miljoenen keren gedownload. Ook Firefox 3.0 (juni 2008) was erg populair en staat zelfs in het Guinness Book



**Afbeelding 1.8** Mozilla Firefox Quantum Developer Edition is een prima browser met een goede implementatie van de webstandaarden.

of Records met het record ‘Meeste downloads in een dag’, ruim 8 miljoen. In 2018 is een nieuwe versie de browser uitgebracht: Firefox Quantum. Begin 2019 is de browser bij versie 65 en updates verschijnen met grote regelmaat, net als bij Chrome. Download Firefox vanaf [www.getfirefox.com](http://www.getfirefox.com).

Net als bij Chrome zijn er veel extensies/add-ons waarmee de functionaliteit van de browser kan worden uitgebreid. Zoek bijvoorbeeld eens met de trefwoorden HTML en CSS naar extensies op [addons.mozilla.org/nl/firefox](http://addons.mozilla.org/nl/firefox).

Van Firefox is ook een versie voor ontwikkelaars beschikbaar: Firefox Quantum Developer Edition. Deze bevat nog meer nuttige hulpmiddelen voor webontwikkelaars dan de gewone editie. Bijzonder handig is de optie om het raster van CSS Grid Layout zichtbaar te maken.

### MacOS: Safari

Alle hiervoor genoemde browsers zijn ook beschikbaar voor het besturingssysteem van Apple: macOS. Dat besturingssysteem heeft echter ook een eigen browser: Safari. De ondersteuning van onderdelen van HTML is vergelijkbaar met Chrome. Bij ontwikkelaars die werken op een Mac is vanwege de goede ondersteuning van standaarden en de uitstekende hulpmiddelen voor ontwikkelaars ook hier Google Chrome een populaire browser. Safari was ooit ook beschikbaar voor Windows, maar dat is niet meer zo. Het marktaandeel op de desktop is bescheiden, maar op mobiel (iPhone) en tablet (iPad) is Safari de meestgebruikte browser.

## HTML-editors

Om webpagina's te kunnen schrijven, hebt u geen bijzondere of dure software nodig. Elke tekstverwerker (*editor*) waarmee tekst als kale ASCII-tekst kan worden opgeslagen, is geschikt. Kladblok (voor Windows) of TekstEditor/SimpleText (voor Mac) voldoet al! Maar voor een comfortabeler leven als webdeveloper zijn er ook tal van gespecialiseerde programma's beschikbaar. Veel ervan zijn gratis.

Gespecialiseerde HTML-editors bieden allerlei voordelen: met codekleuring kunt u makkelijk de verschillende onderdelen zoals tags, attributen, eigenschappen en waarden herkennen, automatisch aanvullen van tags bespaart typewerk, spellingcontrole voor uw code scheelt debugtijd, automatisch inspringen maakt code beter leesbaar en zo is er nog veel meer.

Het aanbod van (gratis) editors is enorm en de keus is persoonlijk. Het beste advies is om er een paar uit te proberen. Verschillende websites bieden toplijst-

ten aan van editors in allerlei soorten en smaken. Kijk voor een neutraal overzicht eens naar [en.wikipedia.org/wiki/List\\_of\\_HTML\\_editors](https://en.wikipedia.org/wiki/List_of_HTML_editors). Bekende namen zijn Visual Studio Code, Brackets ([brackets.io](https://brackets.io)), Atom ([atom.io](https://atom.io)) en Sublime text ([www.sublimetext.com](https://www.sublimetext.com)). Visual Studio Code wordt hierna besproken.



### Microsoft Office is geen codeertool

Gebruik niet de mogelijkheden van Word of Excel om een bestand als HTML-document op te slaan met een menuopdracht zoals **Bestand, Opslaan als webpagina**. Hoewel het resultaat in een moderne versie van Office niet zo rampzalig is als het ooit was, bevat de webpagina veel overbodige code. Wilt u de opmaak van een Word-document behouden, plaats dan een pdf-versie op uw website. Of maak er zelf een HTML-pagina van met CSS voor de opmaak.

---



### CSS-editors

U hebt voor het werken met CSS geen ander programma nodig. Alle moderne editors helpen u bij het schrijven van CSS net zo goed als bij HTML, inclusief automatisch aanvullen van code, de juiste opties voor inspringen, het plaatsen van accolades en codekleuring.

---

## Visual Studio Code

Het overweldigende aanbod van HTML-editors kan het maken een keus nogal bemoeilijken. Vindt u op het eerste gezicht niet iets van uw gading, kijk dan eens naar Visual Studio Code (VS Code), te vinden op [code.visualstudio.com](https://code.visualstudio.com). De editor is gratis en beschikbaar voor Linux, macOS en Windows. Aardigheidje: de interface en alle functies zijn geschreven in HTML, CSS en JavaScript. In feite is het programma een webtoepassing die is omgezet naar een oorspronkelijk Linux-, macOS- of Windows-programma. Dat is overigens meer dan een aardigheidje. Die opbouw zorgt er namelijk voor dat de gebruiker de interface en functies eenvoudig kan aanpassen en dat het betrekkelijk eenvoudig is om er extensies voor te maken. Er zijn dan ook veel extensies beschikbaar.

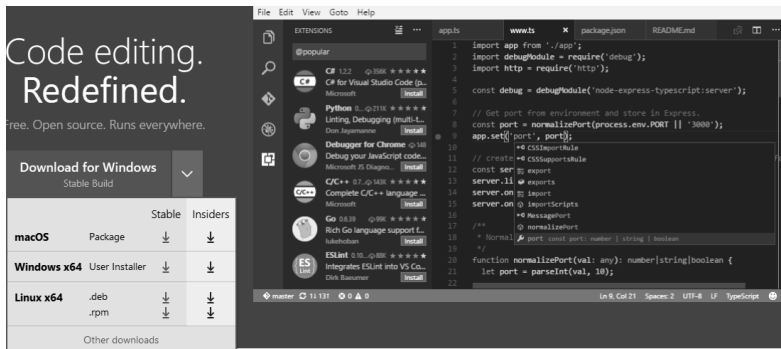


### Electron

Het is wel vroeg om erover te beginnen omdat u nog nauwelijks uit het startblok bent, maar u leest het goed: webtoepassingen kunnen worden omgezet naar programma's die op verschillende besturingssystemen werken. In dit geval is dat gedaan met Electron, een opensourcebibliotheek die wordt ontwikkeld door GitHub. Kijk als u er klaar voor bent eens op [electronjs.org](https://electronjs.org).

---

VS Code is een uitstekende editor voor HTML en CSS, maar ondersteunt veel meer talen, bijvoorbeeld JavaScript, TypeScript, PHP, Java en C++. De basisondersteuning van HTML en CSS is prima, met codesuggesties (IntelliSense), automatisch sluiten van tags, informatie over tags bij aanwijzen, syntaxiskleuring, voorbeeldkleur, code samenvouwen en meer. Ook handig zijn de vele toetsencombinaties voor het bewerken van code: kopiëren van hele blokken, meerdere cursors tegelijk, alle keren dat een woord voorkomt ineens selecteren en meer. Op de website van VS Code staat een overzicht van toetsencombinaties. Met extensies kunt u nog veel meer handigheidjes toevoegen. De website is [marketplace.visualstudio.com/VSCode](https://marketplace.visualstudio.com/VSCode), maar u kunt net zo makkelijk zoeken vanuit de interface van VS Code.

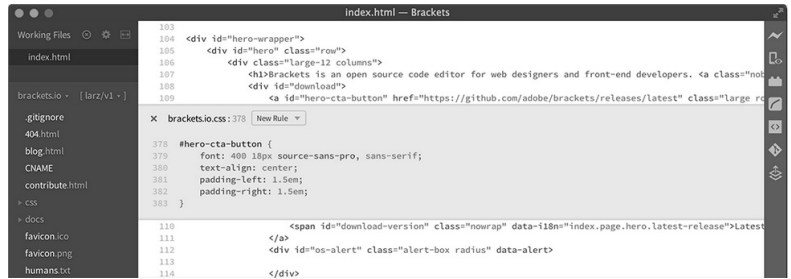


**Afbeelding 1.9** Visual Studio Code, een veelzijdige editor voor HTML en CSS.

### Liveserver

In voorgaande edities van dit boek werd de editor Brackets uitgelicht. Deze editor is niet beter of slechter dan VS Code (of andere editors) en heeft zijn eigen handigheidjes. Zo is het bijvoorbeeld mogelijk om in de HTML-code met een toetsencombinatie direct de bijbehorende CSS-code te bewerken. Een ander pluspunt is de functie Live voorbeeld. Zodra u een HTML- of CSS-bestand wijzigt, ziet u direct in de browser het resultaat. Dit werkt in elk geval in Google Chrome, maar meestal ook in andere browsers. Kies daarvoor de optie **Experimentele versie Live voorbeeld**.

Intussen is voor VS Code een extensie beschikbaar die hetzelfde doet: Live Server. Hiermee wordt op de achtergrond een lokale webserver gestart waarin elke verandering in de code zichtbaar is zodra het document is opgeslagen. En omdat het opslaan kan worden geautomatiseerd, bijvoorbeeld direct als de cursor het codevenster verlaat, werkt dit fantastisch.



Afbeelding 1.10 Brackets.

Al die aandacht voor zoiets als een liveserver lijkt misschien overdreven. Maar zodra u intensief met het schrijven van code aan de slag gaat, zult u merken hoeveel prettiger dat werkt.

## Commerciële editors/IDE's

Naast de gratis editors worden er ook editors tegen betaling aangeboden, hoewel de functionaliteit daarvan veel verder gaat dan alleen HTML- en CSS-code bewerken. Dergelijke producten vallen in de categorie geïntegreerde ontwikkelomgeving: *integrated development environment* of IDE. Ze bieden veel meer mogelijkheden dan u als beginnende ontwikkelaar nodig hebt en zelfs in de fase daarna kunt u met de genoemde gratis editors prima projecten met HTML, CSS en JavaScript uitvoeren. De meerwaarde van de commerciële producten zit vaak in de ondersteuning en nog betere en meer functies voor ontwikkeling. Voorbeelden zijn:

- WebStorm ([www.jetbrains.com/webstorm](http://www.jetbrains.com/webstorm))
- Dreamweaver ([www.adobe.com/nl/products/dreamweaver.html](http://www.adobe.com/nl/products/dreamweaver.html))

## Aanvullende hulpmiddelen

Naast browsers en een code-editor zijn er nog een paar hulpmiddelen waarmee u het schrijven en doorgronden van HTML- en CSS-code vereenvoudigt. In de voorgaande paragrafen zijn er al enkele genoemd, maar voor de duidelijkheid staan ze hier bij elkaar.

### Hulpmiddelen voor validatie

Eén hulpmiddel is nog geheel niet genoemd: de validator, een hulpmiddel om te controleren of uw code aan de webstandaarden voldoet. Een validator controleert of de code voldoet aan de eisen van de taal, zoals complete tags en

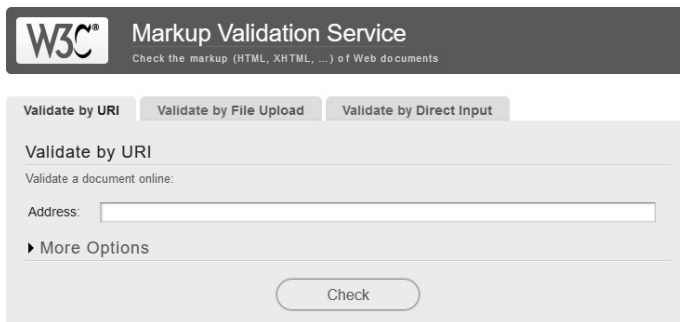
verplichte attributen. U kunt een validator zien als de spellingcontrole van een tekstverwerker. U doet uw best om zo zorgvuldig mogelijk te coderen en de validator wijst u op mogelijk achtergebleven fouten.

Fouten in HTML geven lang niet altijd zichtbare problemen in de browser, maar in complexere webapplicaties kunnen ze de boel flink overhoop gooien. Met een validator bent u zulke problemen voor. Tijdens het leren van HTML is de waarde van een validator dat die laat zien waar uw code beter kan.

Op de website van het W3C zijn validators beschikbaar voor HTML en CSS, en u vindt er links naar andere validators:

- De HTML-validator staat op [validator.w3.org](http://validator.w3.org).
- De CSS-validator is te vinden op [jigsaw.w3.org/css-validator/](http://jigsaw.w3.org/css-validator/).

De werking van de validators wijst zich vanzelf: u kunt een webadres laten controleren, een bestand uploaden of een fragment typen of plakken.

The image shows the W3C Markup Validation Service interface. At the top, there is a dark header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by URI" tab is selected. Underneath, it says "Validate by URI" and "Validate a document online:". There is a text input field labeled "Address:" and a "More Options" link. At the bottom, there is a "Check" button.

**Afbeelding 1.11** HTML-validatieservice van het W3C. Voor CSS is een vergelijkbare interface beschikbaar.

## Hulpmiddelen voor ontwikkeling

Er is een enorme variatie aan hulpmiddelen voor het werken met HTML- en CSS-code. Enkele voorbeelden:

- Hulpmiddelen voor ontwikkelaars in de diverse webbrowsers, eenvoudig te vinden in het menu van de browser. Deze zijn in de loop van de tijd zo uitgebreid dat u hiermee al een goedgevulde gereedschapskist hebt.
- Web Developer Toolbar voor Chrome, Firefox en Opera. Een uitbreiding die als add-on of extensie wordt toegevoegd aan de browser. Deze werkbalk bevat talloze nuttige opties voor werken met CSS en HTML, formulieren, afbeeldingen enzovoort.
- De website [canisuse.com](http://canisuse.com) met ondersteuningstabellen voor webtechnologieën (HTML, CSS, HTML API's).