

Inhoud

1	Webtalen, browsers en editors	1
	Introductie	2
	Een snelle terugblik	2
	Afspraken over webtalen	3
	Twee smaken HTML	5
	Wat HTML5 is	7
	Kennismaken met HTML en CSS	8
	Aan de serverkant	8
	HTML – Hypertext Markup Language	9
	CSS – Cascading Style Sheets	10
	Browsers	11
	Google Chrome	12
	Microsoft Internet Explorer/Edge	13
	Mozilla Firefox	15
	Mac OS X: Safari	16
	HTML-editors	16
	Brackets	17
	Aanvullende hulpmiddelen	19
	Hulpmiddelen voor validatie	19
	Hulpmiddelen voor ontwikkeling	20
	Samenvatting	26
	Oefeningen	26
2	De basis van HTML5	27
	Kenmerken van HTML-documenten	28
	Elementen, tags en tekst	28
	Sluiten in de goede volgorde	31
	De HTML-code van webpagina's bekijken	32
	Attributen	34
	Kenmerken van attributen	35
	Globale attributen	36

Nieuw in HTML5	40
Vervallen onderdelen	43
De basis van een HTML5-document	44
<!DOCTYPE html>	45
<html lang="nl">	45
<head>	45
<meta charset="utf-8">	46
<title>	46
<link>	47
<body>	47
Paginastructuur	47
Document Object Model – HTML DOM	48
De outline van een document	49
Het contentmodel van HTML5	54
Secties markeren – theorie	57
<article> (sectioning content)	59
<section> (sectioning content)	59
<nav> (sectioning content)	60
<aside> (sectioning content)	61
Sectioning root	62
<header> (flow content)	63
<footer> (flow content)	64
<address> (flow content)	66
<main> (flow content)	66
<div> (flow content)	67
Samenvatting	68
Secties markeren – praktijk	68
Eenvoudige webpagina	68
Productenpresentatie	70
Blog	73
Weergave in oude browsers	75
Samenvatting	76
Oefeningen	77
3 Tekst markeren	79
Inleiding	80
Koppen markeren	80
Koppen in een header	81
Tekst markeren	83
Alinea's: het element <p>	83
Het regeleinde 	84
Afbreken van woorden	86

	Speciale betekenissen aangeven	86
	Inhoud groeperen	92
	Vooraf opgemaakte inhoud: <pre>	92
	Citaten gebruiken: <blockquote>	92
	Lijsten maken: , en 	93
	Illustraties en bijschriften: <figure> en <figcaption>	99
	FAQ, metadata en definities: <dl>, <dt> en <dd>	102
	Nieuw thema markeren: <hr>	103
	Speciale tekens in webpagina's	104
	Van ASCII naar UTF-8	105
	Referenties gebruiken	105
	Samenvatting	106
	Oefeningen	106
4	Koppelingen maken	109
	Inleiding	110
	Verbindingen leggen met <a>	111
	Attributen van <a>	112
	Niet volgen: nofollow	115
	Niet klikken: noreferrer	115
	Bladwijzers maken	116
	Link naar een grote foto	117
	Link naar het vervolg van een artikel	118
	Link naar een e-mailadres	119
	Klik hier voor richtlijnen	120
	Externe bestanden koppelen met <link>	122
	Stylesheet koppelen	122
	Een favicon gebruiken	124
	Opeenvolgende pagina's	125
	Vooraf laden: prefetch, prerender en dns-prefetch	126
	Overige waarden van rel	127
	Samenvatting	128
	Oefeningen	128
5	Beeld, geluid en andere externe inhoud	129
	Inleiding	130
	URL's en structuur van de website	130
	Lokale site	132
	Soorten afbeeldingen	132
	Bitmaps	133
	Vectorafbeeldingen: SVG	134

Bestandsformaten voor bitmaps	135
GIF	136
JPEG	137
PNG	138
Afbeeldingen in HTML	139
De bron: src	139
Beschrijvende tekst: alt	139
Afmetingen: width en height	141
Klikbare gebieden: usemap	141
Externe HTML-inhoud: <iframe>	143
Attributen	143
Beveiliging	145
Insluiten met <embed> of <object>	146
Video in een webpagina	147
Bestandstypen voor video	148
Een codec kiezen	148
Video coderen	149
Het element <video>	149
De bron: <source>	151
Hulp voor oudere browsers	152
Ondertiteling met <track>	153
Titelbestanden: WebVTT	154
Audio op de website gebruiken	156
Samenvatting	156
Oefeningen	157
6 Tabellen maken	159
Inleiding	160
De structuur van een HTML-tabel	160
De basis: <table>	162
Rijen en kolommen: <tr> en <td>	164
Koptekst: <thead>	164
Een bijschrift: <caption>	165
Figcaption in plaats van caption	166
Overige elementen	167
De tabelinhoud: <tbody>	167
Voettekst: <tfoot>	167
Kolomgroepen: <colgroup> en <col>	168
Voorbeelden van tabellen	169
Samenvatting	173
Oefeningen	173

7	Formulieren maken	175
	Formulieren in HTML5	176
	Verbeteringen aan de clientkant	176
	De rol van de webserver	177
	De basis van een formulier: <form>	178
	Buiten <form> mag ook	178
	De verwerking: action	178
	Verzendwijze: method	178
	Codering: enctype	179
	Automatisch aanvullen: autocomplete	180
	Niet valideren: novalidate	180
	Direct invoeren: autofocus	180
	De naam: name	181
	Het doel: target	181
	Buitenspelelementen: form	182
	Het element <label>	183
	De tabvolgorde	184
	Het element <input>	184
	Attributen van <input>	185
	Soorten invoer: het attribuut type	191
	Telefoonnummers: <input type="tel">	191
	E-mailadressen: <input type="email">	191
	Webadressen: <input type="url">	192
	Wachtwoorden: <input type="password">	193
	Getallen: <input type="number">	193
	Een bereik: <input type="range">	194
	Kleuren: <input type="color">	195
	Datum en tijd: date en time	196
	Selectievakjes: <input type="checkbox">	198
	Keuzerondjes: <input type="radio">	199
	Bestanden uploaden: <input type="file">	200
	Verzendknop: <input type="submit">	201
	Afbeeldingknop: <input type="image">	201
	Resetknop: <input type="reset">	202
	Functieloze knop: <input type="button">	202
	Verborgene waarden: <input type="hidden">	204
	Het element <button>	204
	Voorbeeld van <button> met afbeelding	205
	Kiezen uit een lijst: <select>, <datalist>, <option>	206
	<option>	206
	<select>	208
	<datalist>	209

Uitgebreide tekstinput: <textarea>	211
Groeperen met <fieldset> en <legend>	212
Voortgang tonen: <progress>	214
Schaal: <meter>	216
Resultaat: <output>	217
Samenvatting	218
Oefeningen	219
8 De basis van CSS	221
Wat CSS is	222
Voorgeschiedenis	222
En toen was er CSS3	223
Waarom CSS zo handig is	224
De taal CSS	225
Waarden en eenheden	227
Lengte	227
CSS-verwerking door de browser	230
De boomstructuur	230
Waarden toewijzen	232
De cascade	232
Specificiteit bereken	233
Overerving	235
De plaats van de CSS-declaraties	236
Het attribuut style	236
Het element <style>	237
Extern bestand: <link>	237
Het CSS-bestand	238
Commentaar toevoegen	238
Andere stijlbladen importen	239
De ingebouwde stijlen aanpassen	239
Elementen benaderen met selectors	240
Universele selector *	241
Typeselector	242
Attribuutselectors	242
Combinators: afstamming, kind en sibling	243
Pseudoklassen	244
Pseudoklasse ontkenning	246
Pseudo-elementen	246
De selectors class en id	248
Best practices	249
Samenvatting	251
Oefeningen	251

9	Boxmodel, weergavemodel en positionering	253
	De opbouw van pagina's	254
	Het boxmodel	254
	De eigenschap box-sizing	255
	Begrensdte breedte en hoogte	258
	Marges	259
	Horizontaal centreren	259
	Verticaal centreren	260
	Negatieve marge	263
	Ingevouwen marges	264
	Padding	265
	Randen	266
	Randdikte	266
	Randkleur	266
	Randstijl	267
	Korte notatie voor randen	268
	Weergavemodel	269
	De eigenschap display	269
	Positionering – de lay-out	271
	Het omvattende blok	272
	Verschuiven met top, right, bottom en left	272
	Relatieve positionering	273
	Absolute positionering	275
	Vaste positie: fixed	276
	Float	276
	Vrije ruimte met clear	278
	Praktijkoplossing: clearfix	279
	Visuele effecten	282
	Overlopende inhoud: overflow	282
	Nieuw: resize	283
	Uitsnijden met clip-path	285
	Zichtbaarheid: visibility	288
	De toekomst	288
	Samenvatting	288
	Oefeningen	289
10	CSS: tekst en typografie	291
	Inleiding	292
	Lettertype: van systeem of online?	293
	Web fonts	293
	Lettertypen downloaden: @font-face	295
	Eigenschappen van @font-face	296
	Een embedcode gebruiken	299

Eigenschappen voor lettertype: font	302
Lettertype	302
Lettergrootte: font-size met px, em en rem	303
Grootte aanpassen: font-size-adjust	303
Regelhoogte: line-height	304
Letterstijl: font-style	304
Lettergewicht: font-weight	305
Uitrekken of indrukken: font-stretch	306
Kleinkapitaal: font-variant	306
De verzameleigenschap font	307
Eigenschappen voor tekst	307
Witruimte behouden: white-space	308
Overlopende tekst: text-overflow	308
Uitlijnen: text-align	309
Inspringen	310
Ruimte tussen woorden en letters	310
Lijneffecten op tekst	311
Tekst omzetten: text-transform	312
Tekst met schaduw	312
Opmaak van lijsten	313
Lijst als navigatie	314
Uitklapbaar menu	315
Samenvatting	317
Oefeningen	318
11 Kleur, randen en achtergronden	319
Inleiding	320
Kleurwaarden	320
RGB en RGBA	321
HSL en HSLA	322
Transparantie	323
Kleur van tekst	324
De achtergrond	325
Achtergrondkleur	325
Achtergrondafbeelding	325
Verskillende achtergrondafbeeldingen	332
Kleurverloop als achtergrond	333
Kleurverlopen	333
Trendgevoelige eigenschap	334
Lineair verloop	334
Radiaal verloop	336
Repeterende verlopen	337
Afgeronde hoeken	338

Randafbeelding	339
Voorbeeld van een randafbeelding	341
Randafbeelding op <body>	342
Schaduw	343
Samenvatting	344
Oefeningen	345
12 Lay-outs maken	347
Inleiding	348
Media queries en responsive design	348
Responsive design en mobile first	351
Weergave op mobiele schermen: de metatag viewport	352
Media queries: breekpunten in de lay-out	354
Vensterbreedte en -hoogte	355
Schermbreedte en -hoogte	356
Schermresolutie	356
Portret of landschap	357
Beeldverhouding	357
Lay-out met positionering en floats	358
Flexibele lay-out met flexbox	366
Flexcontainers en flexitems	367
Het assenstelsel van de flex-flow	368
De richting: flex-direction	369
Regelomloop: flex-wrap	371
De eigenschap flex-flow	372
De volgorde: order	372
Flexitems schalen	373
Instellingen voor flex	375
Flexitems uitlijnen	376
Uitlijnen met de marge	376
Uitlijnen op de hoofd-as: justify-content	376
Uitlijnen op de kruis-as: align-self	378
Regels uitlijnen: align content	378
Een lay-out maken met flexbox	379
Verder ontdekken	384
Samenvatting	384
Oefening	385
Index	387

Webtalen, browsers en editors

HTML, de belangrijkste taal op internet, heeft in de afgelopen jaren een grote ontwikkeling doorgemaakt. De nieuwste afspraken zijn vastgelegd in HTML5. Dit hoofdstuk beschrijft het ontstaan en de actuele situatie. Er wordt kort uitgelegd wat HTML en CSS zijn en wat HTML5 is, want daar wordt verschillend over gedacht. Er is een overzicht van browsers en hulpmiddelen voor het schrijven van HTML-documenten en u maakt kennis met een handige gratis editor.

U leert in dit hoofdstuk:

Hoe we hier gekomen zijn.

De rol van HTML en van CSS, en wat HTML5 is.

Welke browsers niet mogen ontbreken.

Handige hulpmiddelen: editors, validators en developer tools.

Introductie

Elke pagina op het world wide web heeft als basis de taal HTML, de afkorting van Hypertext Markup Language. Het kan er nog zo flitsend en grafisch uitzien en er kunnen allerlei aanvullende talen en technieken zijn gebruikt, de basis, de kern, is altijd HTML. Dit boek gaat over HTML5. Het is de nieuwste versie van de taal, vastgelegd in oktober 2014.

HTML5 kan niet los worden gezien van de voorgangers, HTML 4.01 en XHTML. HTML5 is daar een voortzetting van, waarbij onderdelen zijn aangepast, toegevoegd en afgeschaft, maar alle oude webpagina's blijven werken. Met HTML5 beginnen we dus niet aan een nieuw web. 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. Gebeurt dat niet, dan blijven de webpagina's toch toegankelijk voor bezoekers. Browsers moeten verouderde pagina's blijven weergeven.

Hebt u enige ervaring met HTML, dan zult in dit boek veel elementen van HTML 4 herkennen. Is dit uw eerste kennismaking met HTML, dan leert u – niet gehinderd door soms verouderde kennis – direct hoe het moderne HTML kan worden toegepast.

Het is bijna standaard, maar daarom niet minder relevant, dat in boeken als dit een overzicht van de historie wordt geschetst. We houden het kort. Maar het is bijvoorbeeld handig om te weten dat er verschillende partijen werken aan HTML5.



HTML Space Center

Oké, het HTML Space Center bestaat niet, maar er is wel iets met spaties. De officiële schrijfwijze van HTML5 is zonder spatie. HTML 4 heeft wel een spatie in de naam. Dat ziet er soms wat vreemd of inconsequent uit, maar we volgen daarmee de schrijfwijze van de specificatie op w3.org.

Een snelle terugblik

Toen de eerste editie van dit boek verscheen – voorjaar 2011 – was de inkt van het logo van HTML5 nog nat. In 2014 heeft de beheerder van de webtalen, het World Wide Web Consortium (W3C) de specificatie 'definitief' vastgesteld (in W3C-termen is het nu een *recommendation*). Definitief staat niet voor niets tussen aanhalingstekens, want intussen wordt nog steeds aan de specificatie

gewerkt. De huidige versie is vooral een ijkpunt voor browserbouwers en webontwikkelaars. Wat erin staat, zou ook bruikbaar moeten zijn in moderne browsers. Dit boek beschrijft de situatie zoals die was in het voorjaar van 2016.

Naast de officiële versie van het W3C is er een tweede HTML-specificatie en die gaat door het leven als *HTML Living Standard*. Een updatedatum geeft aan welke versie het betreft.



Afbeelding 1.1 *Het logo van HTML5, in 2011 gepresenteerd door het W3C.*

HTML

Living Standard — Last Updated 3 June 2016

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 HTML een stoornis in de ontwikkeling heeft gehad. Om dat te kunnen begrijpen, moet u iets weten van de geschiedenis.

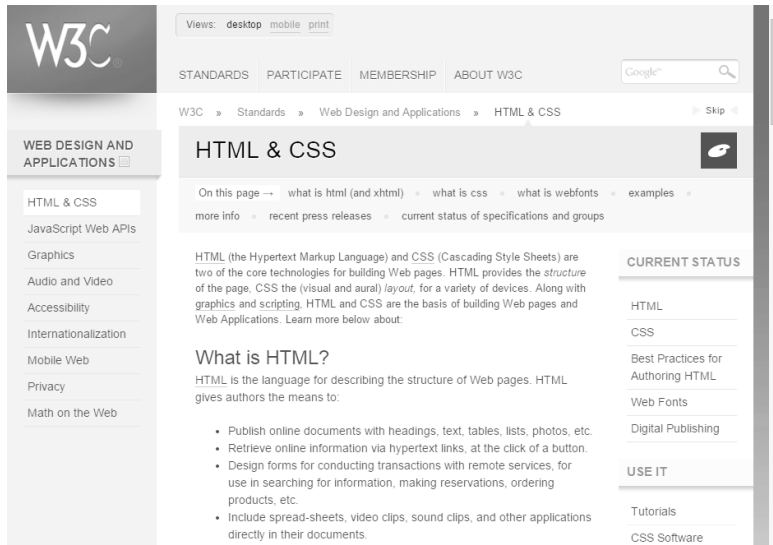
Afspraken over webtalen

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). Daarvoor is het World Wide Web Consortium, kortweg W3C, opgericht. In verschillende werkgroepen overleggen browserfabrikanten zoals Microsoft, Google, Mozilla en Opera. Omdat W3C ook allerlei andere webstandaarden beheert, vindt u onder de leden ook bedrijven als Nokia, Vodafone, Walt Disney Company en Motion Picture Association of America. Het doel van bijvoorbeeld de HTML-werkgroep 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. Dat is de thuishaven van deze in Amerika gevestigde organisatie. U vindt er ook informatie over de hierna genoemde talen XHTML en XML.



Afbeelding 1.3 Het onderdeel HTML en CSS op de website van W3C.

Het is allerminst vanzelfsprekend dat er nu een W3C-standaard van HTML5 is. Eind jaren negentig, de standaard HTML 4.01 was klaar, besloot de organisatie 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 (XHTML2) 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 onvoldoende medestanders voor. In reactie daarop besloten Apple, Mozilla en Opera dan maar zelf aan de slag te gaan met HTML5. 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 technieken zoals Flash van Adobe en Silverlight van Microsoft. Dit initiatief viel bij de gemeenschap van webontwikkelaars in goede aarde. Omdat die de ontwikkelingen met gejuich ontvingen en omdat fabrikanten de nieuwe ideeën gingen inbouwen in hun browsers, ging uiteindelijk ook bij het W3C het roer om. De werkgroep XHTML is eind 2009 opgeheven en de aandacht gaat weer uit naar HTML.



Living Standard

In een dynamische omgeving als internet moeten de mogelijkheden niet beperkt worden door starre regels. Met dat uitgangspunt propageert WHATWG dat HTML een levende standaard is, constant in ontwikkeling. Daar past geen versienummer bij. Lees de FAQ van WHATWG voor achtergrondinformatie over deze organisatie op wiki.whatwg.org/wiki/FAQ.

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. Dit principe staat bekend als Evergreen browsers (zie ook <http://eisenbergeffect.bluespire.com/evergreen-browsers/>).

Twee smaken HTML

Zijn er nu twee smaken HTML? In zeker zin is dat zo. Hoewel de HTML5-recommendation op de website van het W3C het resultaat is van samenwerking tussen W3C en WHATWG, hebben de wegen van de groepen zich gescheiden. Wat W3C betreft is HTML5 'af', hoewel de organisatie sinds 2012 ook werkt aan HTML5.1. Daarbij gaat het om zaken die in de huidige specificatie nog onvolgende ontwikkeld zijn. HTML5.1 is in 2016 een recommendation geworden. In grote lijnen is de HTML5-specificatie van W3C te zien als een stabiel geheel

dat wordt ondersteund door een groot aantal marktpartijen. De Living Standard van WHATWG daarentegen is veel dynamischer. Er wordt permanent gewerkt aan nieuwe of verbeterde mogelijkheden op basis van wat websitebouwers nodig hebben.

Doordat er twee specificaties bestaan, rijst uiteraard de vraag welke u als websitebouwer het best kunt hanteren. Daarbij telt uiteraard welke specificatie door de browserfabrikanten wordt gevolgd en dat is de de W3C-versie. Gebruik die als stabiele standaard om websites te bouwen die door alle moderne browsers worden ondersteund. Volg de 'levende standaard' van WHATWG om te zien wat de toekomst kan brengen.

U hebt als ontwikkelaar in elk geval geen last van de meningsverschillen tussen de specificatiemakers. Integendeel. WHATWG denkt vooral na over wat er mogelijk zou moeten zijn; het zijn de creatieven. Het W3C is een stuk behoudender, maar haalt wel de krenten uit de pap bij WHATWG en moderniseert daarmee de HTML-specificatie die de industrie hanteert. Daar worden de mogelijkheden van HTML alleen maar groter van.



Bekijk de specificaties

Uiteraard zijn alle specificaties online beschikbaar. De W3C-recommendation is te vinden op <http://www.w3.org/TR/html5/>. Deze specificatie toont wat er is. De specificatie van WHATWG is te vinden op www.whatwg.org/html en toont wat er mogelijk komt. Daardoor is de kans ook groter dat er zaken in staan die nog niet in webbrowsers zijn opgenomen. Er bestaat overigens geen overzicht van de verschillen.

W3C Recommendation

W3C

HTML5

A vocabulary and associated APIs for HTML and XHTML

W3C Recommendation 28 October 2014

This Version:
<http://www.w3.org/TR/2014/REC-html5-20141028/>

Latest Published Version:
<http://www.w3.org/TR/html5/>

Latest Version of HTML:
<http://www.w3.org/TR/html/>

Afbeelding 1.4 Hier gaan we het uitgebreid over hebben.

Weliswaar is het getalsmatig een hele stap van 4.01 naar 5, dat betekent niet dat alles van HTML 4 ongeldig en overbodig is nu HTML5 er is. Integendeel. Er zijn wel dingen veranderd, vervallen en toegevoegd, maar het is nog steeds HTML. HTML5 is zogezegd *backward compatible* oftewel uitwisselbaar met voorgaande versies. Hebt u ervaring met HTML, dan kunt u de overstap naar HTML5 beschouwen als bijscholing.



Oude HTML werkt nog

De achterwaartse uitwisselbaarheid (*backward compatibility*) van HTML5 gaat ver. Ook ‘verboden’ opmaaktags zoals `<center>` of `` doen het nog in de allernieuwste browsers. Dat moet ook wel, want er zijn talloze webpagina’s die met deze tags zijn gemaakt en ook die pagina’s moeten toegankelijk blijven voor bezoekers. Wel worden web-sitebouwers aangemoedigd dergelijke tags niet meer te gebruiken. HTML is voor structuur en inhoud, niet voor opmaak. Verderop wordt dit uitgelegd.

Wat HTML5 is

HTML5 is in essentie niet meer dan een upgrade van HTML 4. We hebben het dan alleen over de markeertaal HTML. HTML5 is ook het modewoord of de marketingterm geworden voor een verzameling van webtechnieken om moderne (mobiele) webapplicaties te bouwen. Naast HTML gaat het dan ook over technieken zoals CSS, JavaScript, XML, JSON en SVG. Bedenk dat deze geen deel uitmaken van de HTML5-standaard van W3C. U zult ze daar niet in terugvinden. Dat neemt niet weg dat ze in het leven van de webontwikkelaar een onmisbare rol spelen. Een korte toelichting:

- CSS staat voor Cascading Style Sheets en is de taal voor de vormgeving van HTML-pagina’s. CSS komt in dit boek uitgebreid aan bod.
- JavaScript is een programmeertaal (scripttaal) om onder meer webpagina’s interactief te maken. JavaScript wordt in dit boek niet behandeld.
- XML staat voor Extensible Markup Language en is een markeringstaal (zoals HTML). Het 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 alternatief voor XML. De functie is hetzelfde: uitwisseling van gegevens tussen een server 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.

Kennismaken met HTML en CSS

De volgende paragrafen veronderstellen dat HTML en het maken van webpagina's voor u volledig onbekend terrein zijn. Hebt u al ervaring met HTML en CSS, dan kunt u doorbladeren naar de paragrafen *Browsers* en *HTML-editors*.

Dit boek gaat allereerst over HTML. Maar tenzij u kale pagina'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 gecodeerde teksten (en afbeeldingen en meer) eruit moeten zien. Ook CSS komt daarom aan bod.

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 of Aurelia. Het is zeker een goed idee om u ook in JavaScript te verdiepen als u verder wilt met webontwikkeling.

HTML, CSS en ook JavaScript doen hun werk 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. Dan vermoedt u dat er waarschijnlijk ook een ander kant is; dat is ook zo.

Aan de serverkant

Er zijn ook talen en technieken waarbij het werk op de webserver wordt gedaan, 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 (kortweg cms) zoals WordPress (www.wordpress.org) of Joomla! (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 kunt u er echter van alles over vinden.

HTML – Hypertext Markup Language

HTML staat voor Hypertext Markup Language. Met HTML wordt de *structuur* van de pagina en de *betekenis* van de elementen in die pagina aangegeven. (In Engelstalige literatuur wordt gesproken over *structure* en *semantics*; onthoud deze termen). 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 daarom geen programmeertaal. Het is een markeertaal. Vandaar ook de afkorting Hypertext Markup Language, oftewel 'een markeertaal die is gebaseerd op hypertext'. Het principe van hypertext kent ondertussen iedereen. Dit zijn de tekstdelen waarmee relaties (ook wel koppelingen of links) tussen de pagina's onderling zijn aangegeven. Vrijwel elke webpagina bevat wel een of meer hyperlinks.

De hyperlinks, maar ook alle andere kenmerken, 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 hoe ze worden gemarkeerd. Stel dat u een stuk tekst op een pagina als koptekst wilt aanmerken. De code hiervoor is:

```
<h1>Dit is een koptekst</h1>
```

De code `<h1>` is de openingstag. Alles wat hierna volgt (in dit voorbeeld de tekst 'Dit is een koptekst') wordt als koptekst aangemerkt. Dit gaat zo door tot dat de sluittag `</h1>` wordt geschreven. Een stuk gemarkeerde tekst wordt altijd afgesloten met een sluittag. Een sluittag heeft dezelfde naam als de openingstag, maar begint met een voorwaartse slash `</...>`.

In de browser ziet de koptekst er zo uit als in de afbeelding. HTML bevat meer dan honderd elementen. In dit boek worden de meeste ervan besproken.

Dit is een koptekst

Dit is de code voor de koptekst:

```
<h1>Dit is een koptekst</h1>
```

Afbeelding 1.5 Een koptekst met daaronder de HTML-code.

CSS – Cascading Style Sheets

Met CSS wordt het uiterlijk van de pagina aangegeven. CSS-code beschrijft hoe de kopteksten, alineateksten en afbeeldingen worden opgemaakt. In CSS wordt aangegeven welk lettertype wordt gebruikt, welke lettergrootte, letterkleur, uitlijning, regelafstand, witruimte tot andere onderdelen en meer. Met CSS is ook de opmaak van de pagina in te stellen in kolommen, kop- en voetteksten en kaders. CSS heeft niets te maken met de inhoud van de pagina.

Neem nog even het voorbeeld van de koptekst uit de voorgaande paragraaf in gedachten. In HTML hebt u aangegeven dat een bepaalde tekst een koptekst moet zijn, maar dat zegt nog niets over het uiterlijk. Wilt u de koptekst bijvoorbeeld in een ander lettertype dan de gewone tekst, met een lijn eronder of nog anders? Dit is nu juist de taak van CSS. Onthoud de volgende uitspraken:

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

Om de koptekst uit de afbeelding 1.3 te voorzien van een blauwe achtergrond met donkerblauwe onderrand en een cursief lettertype, kan de volgende stylesheet worden geschreven:

```
h1 {  
  font-family: Arial, Helvetica, sans-serif;  
  font-style: italic;  
  background-color: hsl(200, 85%, 50%);  
  border-bottom-width: 4px;  
  border-bottom-style: solid;  
  border-bottom-color: hsl(200, 85%, 35%);  
  padding-left: 15px;  
}
```

De complete CSS-familie bestaat ook uit enkele honderden opdrachten die gezamenlijk alle mogelijke manieren bieden om een webpagina op te maken. Hoe meer CSS-regels u kunt toepassen, hoe fraaier het eindresultaat in de browser kan zijn.

Dit is een koptekst

```
h1 {
    font-family: Arial, Helvetica, sans-serif;
    font-style: italic;
    background-color: hsl(200, 85%, 50%);
    border-bottom-width: 4px;
    border-bottom-style: solid;
    border-bottom-color: hsl(200, 85%, 35%);
    padding-left: 15px;
}
```

Afbeelding 1.6 Een koptekst opgemaakt met een stylesheet. In de HTML-code is niets gewijzigd, alleen het uiterlijk is aangepast met een stijlblad.



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. De specificatie bestaat uit losse modules, is zeer omvangrijk en (daardoor) nog lang niet af. Verschillende onderdelen (modules) zijn wel bruikbaar en dat worden er steeds meer. Deze komt u tegen in de hoofdstukken over CSS.

Browsers

Om webpagina's te kunnen maken en de uitleg in dit boek goed te kunnen volgen, hebt u enkele programma's nodig op uw computer. Het maakt daarbij niet uit of u werkt op een pc met Windows of Linux of op een Mac met Mac OS X (al zullen de gebruikte programma's natuurlijk iets verschillen). Alle voorbeelden zijn geschikt voor alle platforms. De code in dit boek is *platformonafhankelijk*. Vaak zullen de benodigde programma's al zijn geïnstalleerd op uw computer, maar voor de volledigheid krijgt u een kort overzicht.

Een van de belangrijkste programma's is natuurlijk de browser. Tijdens het ontwerpen van webpagina's gebruikt u de browser om te kunnen zien hoe de eindgebruikers uw pagina's te zien krijgen. Het liefst beschikt u zelfs over twee of meer browsers om verschillen te kunnen ontdekken. Ziet uw ontwerp er in de ene browser heel anders uit dan in de andere, dan zult u de code moeten aanpassen. Dat zal overigens zelden de HTML-code zijn (HTML gaat over betekenis, niet over uiterlijk), maar meestal de CSS-code (CSS is voor de opmaak).

De drie meest gebruikte browsers zijn Chrome, Internet Explorer en Firefox. Hoewel het lastig is om exacte marktaandeelen te achterhalen, is de grote lijn duidelijk. Chrome heeft een (stijgend) aandeel van circa 50%, Internet Explorer/Edge, Firefox en Safari schommelen rond de 15% en Opera staat op circa 5%. Het aandeel van Google Chrome groeit niet alleen dankzij de populariteit op desktops, maar ook doordat Android (ook Google) zo goed boert in de mobiele markt. Die is ook belangrijk voor Safari (iPhone/iPad).



Ondersteuning voor HTML5

Of een browser HTML5 programmatisch ondersteunt, is niet voor alle vernieuwingen van belang. Zolang het gaat om elementen die de documentstructuur bepalen, hoeft een browser alleen maar te weten dat het element er is, zodat u het kunt opmaken met CSS. De meeste elementen ‘doen’ namelijk niets. Het ligt anders bij onderdelen van HTML5 waarbij de webbrowser de inhoud van elementen moet vertalen in een handeling, bijvoorbeeld het vertonen van een filmpje op de plaats van de tag `<video>` of het valideren van een formulierveld. Dit wordt verderop in dit boek uitgelegd.



Verschillen tussen browsers

Het is nogal een open deur, maar uw website moet in alle browsers goed werken en het bedoelde ontwerp zo goed mogelijk weergegeven. U zult hierbij heel wat minder problemen tegenkomen dan enkele jaren geleden gebruikelijk was, want browsers werken steeds beter volgens de webstandaarden. Het wordt een heel ander verhaal als uw site ook moet werken in antieke browsers (probeer dat te vermijden!) of bij het bouwen van geavanceerde webapplicaties. Voor dergelijke situaties zijn bijvoorbeeld JavaScript-bibliotheken beschikbaar die testen of de browser een bepaalde functie ondersteunt en u de mogelijkheid geven een alternatief in te bouwen. Dergelijke geavanceerde technieken worden in dit boek niet behandeld.

Google Chrome

Het lijkt alsof hij er altijd al was, maar toch is Chrome, de browser van Google, pas eind 2008 gelanceerd. Chrome lijkt intussen de strijd met Internet Explorer en Firefox te hebben beslist. Vanwege het grote marktaandeel mag deze browser niet op uw computer ontbreken. U kunt hem zelf downloaden en installeren vanaf www.google.com/chrome. Chrome geeft de webstandaarden goed weer. De browsers was aanvankelijk gebouwd op de engine WebKit, die ook in Safari van Apple wordt gebruikt. Sinds april 2013 bevat Chrome een eigen versie van WebKit genaamd Blink.