

Inhoud

1	Kennismaken met HTML5	1
	Een korte geschiedenis van HTML	2
	Wie heeft het web bedacht?	3
	Afspraken over webtalen	4
	WHATWG	5
	Twee smaken HTML	6
	Backward compatible	7
	Wat HTML is	8
	HTML bestaat uit tags	9
	Wat behoort niet tot HTML5?	10
	Een kort woordje over CSS	11
	CSS in dit boek	13
	Clients en servers	13
	Browsers	15
	Google Chrome	16
	Microsoft Internet Explorer	17
	Mozilla Firefox	17
	Mac OS X: Safari	19
	HTML-editors	19
	Voorbeelden van editors	21
	Brackets	21
	Aanvullende hulpmiddelen	24
	Hulpmiddelen voor validatie	24
	Hulpmiddelen voor ontwikkeling	25
	Chrome Developer Tools	26
	Verder benodigde voorkennis	31
	Tips voor meer lezen	32
	Oefenbestanden downloaden	33
	Samenvatting	34
	Praktijkoefeningen	34

2	Beginnen met HTML5	39
	Theorie 1 – kenmerken van HTML-documenten	40
	Kale tekst	40
	HTML-bestanden heten .html	40
	Theorie 2 – elementen, tags en tekst	41
	Punthaken < en >	43
	Sluittag </...>	43
	Elementen binnen elementen: nesting	44
	Elementen in kleine letters	45
	Praktijk – HTML-code van webpagina's bekijken	46
	Theorie 3 – attributen	49
	Kenmerken van attributen	49
	Praktijk – de basis van elk HTML5-document	51
	<!DOCTYPE html>	53
	<html lang="nl">	53
	<head>	54
	<meta charset="utf-8">	54
	<title>	55
	<body>	56
	HTML-bestand uitbreiden met <link>	56
	Praktijk – de pagina verder vullen	57
	Kopteksten gebruiken	57
	Attributen voor kopteksten	60
	Teksten en alinea's toevoegen	61
	Enter, spaties en tabs worden genegeerd	61
	Alinea's aangeven met <p>	62
	Attributen voor <p>	63
	Het regeleinde 	63
	Afbreken van woorden	64
	Afbeeldingen invoegen	65
	De tag 	65
	Conclusie	68
	Praktijkoefeningen	69
	Uw website	69
	Andere oefeningen	70

3	HTML5-structuurtags	73
	Meer over paginastructuur	74
	Een 'website als boek'	74
	Document Object Model – HTML DOM	74
	Bij foutieve code...	76
	De outline van een document	77
	Kopteksten en outlines	79
	Nieuwe vertakking beginnen	80
	Secties invoegen	81
	Het contentmodel van HTML5	83
	Het contentmodel samengevat	85
	Secties markeren – theorie	86
	Markeringen: vroeger	86
	Markeringen: nu	88
	<article> (sectioning content)	89
	<section> (sectioning content)	90
	<nav> (sectioning content)	91
	<aside> (sectioning content)	92
	Sectioning root	93
	<header> (flow content)	93
	<footer> (flow content)	95
	<address> (flow content)	96
	<main> (flow content)	97
	<div> (flow content)	97
	Samenvatting	98
	Secties markeren – praktijk	98
	Voorbeeld 1 – eenvoudige webpagina	99
	Voorbeeld 2 – productpresentatie	101
	Voorbeeld 3 – een blog	105
	Samenvatting	108
	Weergave in oude browsers	108
	Vervallen in HTML5	109
	Conclusie	110
	Praktijkoefeningen	111
	Uw website	111
	Andere oefeningen	112

4	Lijsten, speciale tekens en andere tekstmarkeringen	115
	Inleiding	116
	Meer over markeren	116
	Koppen in een header	116
	Speciale betekenis aangeven met inline elementen	117
		118
		119
	<small>	119
	<s>	120
	<cite> en <q>	120
	<abbr> en <dfn>	120
	<code>, <samp> en <kbd>	121
	<sub> en <sup>	122
	<mark>	122
		123
	<ins> en 	123
	Vooraf opgemaakte inhoud: <pre>	124
	Lijsten maken: , en 	124
	Genummerde lijsten:	125
	Ongenummerde lijsten:	128
	Lijsten als navigatiemenu	130
	Kop boven een lijst	130
	Illustraties en bijschriften: <figure> en <figcaption>	131
	Nieuw thema markeren: <hr>	133
	Speciale tekens in webpagina's	134
	Named character reference	135
	Van ASCII naar UTF-8	136
	Unicode	136
	Referenties gebruiken	136
	Conclusie	138
	Praktijkoefeningen	139
	Uw website	139
	Andere oefeningen	139

5	Meer over koppelingen	141
	Inleiding	142
	Verbindingen leggen met <a>	143
	Attributen van <a>	144
	Het attribuut title	146
	De attributen target en download	146
	Bestanden downloaden vanaf website	148
	Niet volgen: nofollow	148
	Bladwijzers maken	149
	Link naar een grote foto	150
	Link naar het vervolg van een artikel	151
	Link naar een e-mailadres	153
	Klik hier voor richtlijnen	154
	Geen 'klik hier'	155
	Absolute en relatieve koppelingen	157
	Absolute koppeling	157
	Relatieve koppeling	157
	Koppeling naar bestanden in andere map	158
	Externe bestanden koppelen met <link>	160
	Stylesheet koppelen	160
	Een favicon gebruiken	161
	favicon.ico	162
	Opeenvolgende pagina's aangeven met rel="..."	164
	Overige waarden van rel	164
	Koppelingen naar dynamische webpagina's begrijpen	165
	Parameters doorgeven	166
	Voorbeeld	167
	Conclusie	168
	Praktijkoefeningen	169
	Uw website	169
	Andere oefeningen	170

6	Beeld, geluid en andere externe inhoud	171
	Inleiding	172
	Embedded content	172
	URL's en structuur van de website	172
	Soorten afbeeldingen	173
	Bitmaps	174
	RGB-model	175
	Vectorafbeeldingen: SVG	175
	Bestandsformaten voor bitmaps	178
	GIF	178
	JPEG	179
	PNG	181
	Afbeeldingen in HTML	182
	De bron: src	183
	Beschrijvende tekst: alt	183
	Afmetingen: width en height	185
	Externe HTML-inhoud: <iframe>	186
	Attributen	186
	Sandbox-beveiliging	188
	Insluiten met <embed> of <object>	189
	Video in een webpagina	191
	Bestandstypen voor video	192
	Een codec kiezen	193
	Video coderen	194
	Het element <video>	194
	Attributen voor <video>	194
	Voorbeeld van <video>	195
	De bron: <source>	196
	Hulp voor oudere browsers	198
	Ondertiteling met <track>	199
	Titelbestanden: WebVTT	202
	Audio op de website gebruiken	204
	Conclusie	205
	Praktijkoefeningen	206
	Uw website	206
	Andere oefeningen	207

7	Tabellen en formulieren	209
	Inleiding	210
	De structuur van een HTML-tabel	210
	De basis: <table>	212
	Attributen van <table>	214
	Rijen en kolommen: <tr> en <td>	215
	Koptekst: <thead>	216
	Een bijschrift: <caption>	217
	Overige elementen	218
	De tabelinhoud: <tbody>	218
	Voettekst: <tfoot>	219
	Voorbeelden van tabellen	219
	Producttabel	219
	Tabel met gecombineerde rijen en kolommen	221
	Formulieren maken – Inleiding	223
	Verbeteringen aan de clientkant	224
	De rol van de webserver	226
	De basis van een formulier: <form>	226
	Buiten <form> mag ook	226
	De verwerking: action	227
	Verzendwijze: method	227
	Codering: enctype	228
	Automatisch aanvullen: autocomplete	228
	Niet valideren: novalidate	229
	Direct invoeren: autofocus	229
	De naam: name	230
	Het doel: target	231
	Een voorbeeldformulier	231
	Het element <label>	231
	Invoervelden met <input>	233
	Attributen van <input>	234
	Onmisbaar: name	234
	value	235
	pattern	236
	size	236
	readonly en disabled	237
	required	238
	placeholder	239
	min en max	239

Meer typen invoervelden	240
Telefoonnummers: <code><input type="tel"></code>	240
E-mailadressen: <code><input type="email"></code>	242
Webadressen: <code><input type="url"></code>	242
Wachtwoorden: <code><input type="password"></code>	243
Getallen, bereik, datum, tijd en kleuren	243
Selectievakjes: <code><input type="checkbox"></code>	244
Keuzerondjes: <code><input type="radio"></code>	247
Knoppen – verzendknop: <code><input type="submit"></code>	248
Resetknop: <code><input type="reset"></code>	248
Functieloze knop: <code><input type="button"></code>	249
Verborgene waarden: <code><input type="hidden"></code>	249
Selectielijsten met <code><select></code>	250
<code><option></code> en <code><optgroup></code>	250
Uitgebreide tekstinput: <code><textarea></code>	252
Het formulier verzenden	253
Formulierscripts op internet	254
Standaardscripts	255
Testscript bij dit boek	255
Oefening – Een formulier maken en verzenden	256
Scripts bij uw provider	258
Conclusie	259
Praktijkoefeningen	260
Uw website	260
8 HTML5 API's gebruiken	263
Inleiding – welke API's zijn beschikbaar?	264
Webapps	264
API – Application Programming Interface	264
Geïntegreerde API's	265
Gerelateerde API's	266
API's in dit hoofdstuk	267
Let op: echt programmeren!	267
Video zelf besturen	268
Stappenplan	269
Stap 1 – de HTML-code	269
Stap 2 – JavaScriptcode schrijven	270
Analyse	270
Stap 3 – continu afspelen door event af te vangen	271
Stap 4 – tijdcode weergeven	273
Meer opties voor video	274

De API Web Storage	275
Korte geschiedenis van lokale opslag	275
Beperkingen van cookies	276
HTML5 Storage en de kenmerken ervan	277
Naam-waardeparen	277
Opslagruimte	277
window.localStorage	278
localStorage gebruiken	279
Enkele Storage-voorbeelden	280
Een paginateller	281
Formuliergegevens in localStorage	283
Meer over localStorage	286
De API Geolocation	287
Kenmerken	288
Geolocation in combinatie met Google Maps	289
Meer HTML5 API's	293
Conclusie	295
Praktijkoefeningen	297
 Index	 317

Kennismaken met HTML5

HTML, de belangrijkste taal op internet, heeft in ongeveer vijf jaar een enorme 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. U ziet wat de plek van dit boek is binnen de serie Web Development Library en leest welke tools nodig zijn om dit boek zo optimaal mogelijk te gebruiken.

In dit hoofdstuk:

Een korte geschiedenis van HTML en HTML5.

De rol van HTML, CSS en verwante webtalen.

Browsers en editors.

Indeling van dit boek.

Praktijkoefeningen.

Een korte geschiedenis van HTML

Elke pagina op het world wide web heeft als basis de taal HTML. Dit is de afkorting van *HyperText Markup Language*. Een site kan er nog zo flitsend uitzien en er kunnen allerlei aanvullende technieken zijn gebruikt, maar de basis 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 daarvan een voortzetting. Er zijn onderdelen aangepast, toegevoegd en afgeschaft, maar alle oude webpagina's blijven werken. Met HTML5 beginnen we dus niet aan een compleet nieuw web. 'Afschaffen' betekent in dit verband dan ook vooral dat auteurs worden aangemoedigd websites volgens de 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 kunnen blijven weergeven.

Hebt u al 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.



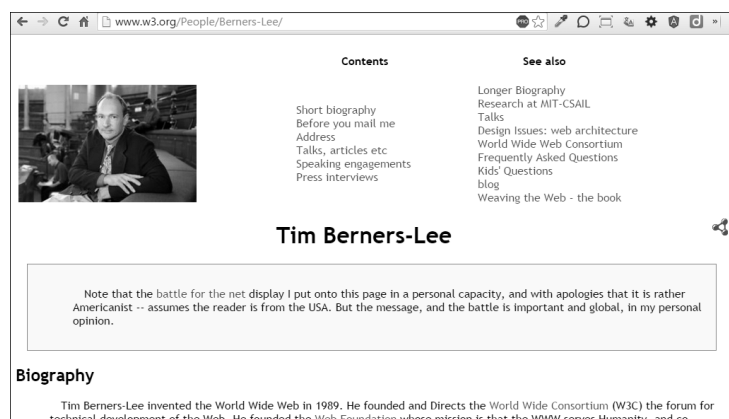
HTML 4 en HTML5

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 **w3.org**.

Wie heeft het web bedacht?

Het is in deze tijd van multinationals en ontwikkelteams die uit tientallen of honderden personen bestaan bijna niet meer voor te stellen. Maar HTML, het web, de webserver en de browser zijn in principe bedacht door één man, Tim Berners-Lee. Hij werkte rond 1990 bij het nucleaire instituut CERN in Zwitserland en zocht naar een manier om documentatie binnen het bedrijf makkelijk en centraal op te slaan. Via onderlinge relaties (de *hyperlinks*) konden hierbij steeds gerelateerde documenten worden bereikt.

Simpel gezegd is het web zoals we dat nu kennen met alle websites, e-commerce en streaming video en audio oorspronkelijk dus ontstaan als een eenvoudig elektronisch documentbeheersysteem.



Afbeelding 1.1 *Tim Berners-Lee, de man die HTML en het web bedacht heeft, is nu nog steeds voorzitter van W3C.*

Tim Berners-Lee is in zijn huidige functie als algemeen directeur van het World Wide Web Consortium W3C nog steeds betrokken bij de ontwikkeling van het web. Als u meer inzicht wilt in de achtergronden van het ontstaan van het web of de manier waarop Berners-Lee het web bedoeld heeft, lees dan ook zijn boek

Weaving the Web (ISBN 006251587X) en bekijk zijn persoonlijke homepage op de site van W3C. Hier staan links naar interessante artikelen en krijgt u inzicht in het ‘ideaalplaatje’ voor het world wide web.

Tim Berners-Lee heeft HTML en zijn serversoftware destijds vrijgegeven als open source. Er gingen veel andere bedrijven mee aan de slag, waaronder Netscape, Microsoft en andere pioniers op internetgebied. Dit had echter ook wildgroei tot gevolg. Iedereen breidde de HTML-‘standaard’ uit met eigen mogelijkheden.

Afspraken over webtalen

Sinds 1994 wordt geprobeerd lijn te brengen in de ontwikkeling van 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, maar ook telecombedrijven en andere industrieën.



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.

In de loop der jaren zijn er allerlei versienummers voor HTML geweest, variërend van HTML 2.0 via 3.2 naar HTML 4.0 en 4.01. Daarna is nog een aantal jaren de variant XHTML gepromoot (dat was HTML gebaseerd op XML), maar die ontwikkeling is in 2009 gestopt. Sindsdien gaat de standaard als HTML5 door het leven.

WHATWG

Sinds 2004 is er ook een alternatieve groepering aan het werk met HTML5, aangevoerd door Apple, Mozilla en Opera. Dit is de *Web Hypertext Application Technology Working Group*, kortweg WHATWG. Zij publiceren eigen modules en standaarden rondom HTML5. Ze noemen dit de *HTML Living Standard*.



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. Met name Google (Chrome), maar ook Mozilla (Firefox) en Opera 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 internet-publiek. 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'. Sinds 2012 wordt daarom gewerkt aan HTML5.1. Daarbij gaat het om zaken die in de huidige specificatie nog onvolgende ontwikkeld zijn. 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.



Afbeelding 1.2 De logo's van HTML5 en HTML Living Standard.

Voor uw leerproces in dit boek maakt het niet uit dat er in de kleine lettertjes verschillen bestaan tussen HTML5 en HTML Living Standard. Alle voorbeelden en codes werken in alle browsers.



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 web-browsers zijn opgenomen.

Backward compatible

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.



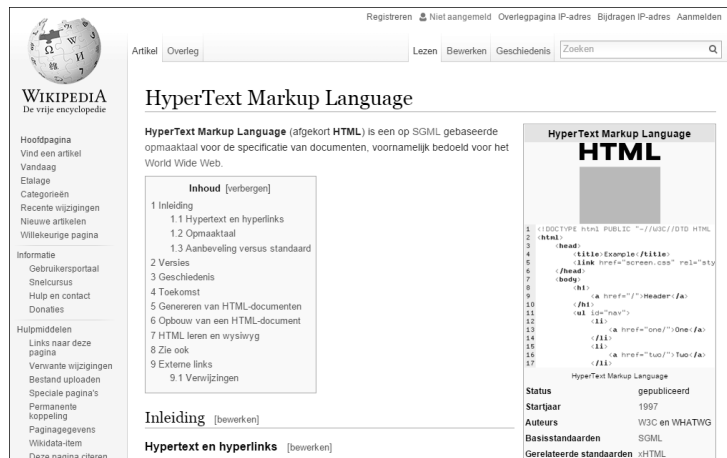
Oude HTML werkt nog

De backward compatibility van HTML5 gaat ver. Ook ‘verboden’ opmaaktags zoals `<center>` of `<font>` 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 websitebouwers aangemoedigd dergelijke tags niet meer te gebruiken. HTML is voor structuur en inhoud, niet voor opmaak. Verderop wordt dit uitgelegd.

Wat HTML is

Met HTML wordt de *structuur* van een webpagina aangegeven. Er wordt in code geschreven wat de kopteksten en alineateksten van een pagina zijn, welke afbeeldingen in de pagina moeten worden geladen en meer. Via hyperlinks wordt aangegeven wat de onderlinge relatie tussen pagina’s is. HTML heeft niets te maken met het uiterlijk van een pagina. In HTML worden alleen de inhoud en de structuur beschreven.

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.



Afbeelding 1.3 *Gestructureerde tekst en hyperlinks vormen de basis van het web.*

HTML5 is een upgrade van HTML 4. We hebben het dan alleen over de markeertaal HTML. Niet over technieken die vaak in één adem met HTML worden genoemd, maar er in feite niet zo veel mee te maken hebben (zie ook verderop).

HTML bestaat uit tags

De hyperlinks, maar ook alle andere structuurkenmerken worden aangegeven met *tags*. Om HTML te leren moet u weten welke tags er zijn en hoe ze genoteerd worden. Stel dat u bijvoorbeeld een stuk tekst op een pagina als koptekst wilt aanmerken. De code hiervoor is dan:

```
<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 aange-merkt. Dit gaat zo door totdat 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. In totaal kent HTML meer dan honderd verschillende tags. In dit boek leert u de belangrijkste en meestgebruikte.



Afbeelding 1.4 Een niet-opgemaakte HTML-koptekst in de browser.

Wat behoort niet tot HTML5?

HTML5 is ondertussen ook een modewoord of 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. De volgende paragraaf gaat kort in op CSS, maar in dit boek staan we er verder niet uitgebreid bij stil. Lees bijvoorbeeld *Web Development Library – CSS3* (ISBN 9789059408081) als u meer wilt leren over CSS.
- JavaScript is een programmeertaal (scripttaal) om onder meer webpagina's interactief te maken. JavaScript wordt in dit boek niet besproken, al komt het in hoofdstuk 8 wel ter sprake. Ook hiervoor is een aparte uitgave verschenen, *Web Development Library – JavaScript* (ISBN 9789059407589).

- 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. XML hebt u vrijwel niet meer nodig als u op het web aan de slag gaat.
- 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 in modern webdevelopment vaak gebruikt in combinatie met JavaScript. Voor HTML is JSON bijna niet van belang.
- 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). Het komt kort ter sprake in hoofdstuk 6.

Een kort woordje over CSS

HTML zegt niks over de vormgeving van de pagina. Daar is CSS voor ontwikkeld. 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 nog veel meer. Met CSS is ook de lay-out 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 vorige afbeelding te voorzien van een blauwe achtergrond met donkerblauwe onderrand en een cursief lettertype, kan bijvoorbeeld 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;  
}
```



Afbeelding 1.5 Een koptekst opgemaakt met een stylesheet. In de HTML-code is niets gewijzigd, alleen het uiterlijk is aangepast CSS.

De complete CSS-familie bestaat 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.