

Basis HTML

In dit hoofdstuk maken we kennis met de ontwikkeling van webpagina's. We gaan dieper in op de structuur en opbouw van een eenvoudige HTML5-pagina en bekijken enkele belangrijke HTML5-elementen.

HTML5 is (sinds 2014) de nieuwste versie van de HTML-standaard en bevat alle functionaliteiten van zowel HTML als XHTML. In deze markeertaal werden enkele foutjes uit de voorgaande versie weggewerkt en een betere ondersteuning geïmplementeerd voor webapplicaties. Daarnaast introduceert HTML5 ook nieuwe elementen die zorgen voor meer structuur in het document.

Starten met HTML

Een lege webpagina bevat onderstaande basis.

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <meta charset="utf-8">
    <title>Dit is de titel van de webpagina</title>
  </head>
  <body>
  </body>
</html>
```

- De `<!DOCTYPE html>` laat de browser weten welke versie (= HTML5) we gebruiken.
- Het `<html>`-element representeert de root van een HTML-document.
- Het `<head>`-element bevat alle informatie die betrekking heeft op het document, maar niet in het documentvenster wordt weergegeven. Het bevat:
 - alle `<meta>`-elementen met informatie voor de browser en zoekmachines.
 - een `<title>`-element
 - de linken naar de stijlbladen en javascriptbestanden.
- Het `<body>`-element bevat alle verdere teksten, afbeeldingen en codes.

Elementen en tags

Aan een lege webpagina kunnen we HTML-elementen toevoegen.

HTML-elementen mét inhoud hebben een begin- en eindtag. Deze tags noemen we 'containertags'. Tussen de begintag en de eindtag staat de inhoud waarop het element betrekking heeft.

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <meta charset="utf-8">
    <title>Dit is de titel van de webpagina</title>
  </head>
  <body>
    <p>Dit is een paragraaf.</p>
  </body>
</html>
```

Sommige elementen zijn 'void elementen' en hebben geen inhoud. Deze elementen hebben hierdoor ook geen eindtag. Deze tags noemen we 'lege tags'.

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <meta charset="utf-8">
    <title>Dit is de titel van de webpagina</title>
  </head>
  <body>
    <p>Dit is een <br /> paragraaf.</p>
    
  </body>
</html>
```

Elementen nesten

Een HTML-element kan andere HTML-elementen bevatten. Dit noemen we 'nesten'. Tijdens het schrijven van onze code, gaan we de elementen laten inspringen om dit weer te geven. Hierdoor krijgen we een duidelijk beeld van het child-element en het parent-element.

Een element kan tegelijkertijd zowel een 'parent' als een 'child' zijn.

Attributen

Elk HTML-element kan in de begintag attributen meekrijgen. Attributen voorzien het element van extra eigenschappen en kenmerken. Er bestaan verschillende soorten attributen:

- *Globale attributen:* attributen die op alle HTML-elementen toepasbaar zijn.
- *Specifieke attributen:* attributen die op specifieke HTML-elementen toepasbaar zijn.

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <meta charset="utf-8">
    <title>Dit is de titel van de webpagina</title>
  </head>
  <body>
    <p class="par" style="color:red">Dit is een paragraaf.</p>
    <a href="http://www.pxl.be">Link naar de PXL-website</a>
  </body>
</html>
```

OPGELET! In het volgende hoofdstuk leren we hoe we de inhoud (HTML) van de opmaak (CSS) scheiden, vanaf dan mogen we het attribuut 'style' niet meer gebruiken. Alle attributen moeten dan een inhoudelijke betekenis hebben!

Commentaar

Soms kan het handig zijn om commentaar toe te voegen over onze code.

```
<!--Dit is een stukje commentaar-->
```

Conventies en validatie

Om ervoor te zorgen dat we een 'goede', 'uniforme', en 'browseronafhankelijke' HTML-code schrijven, houden we ons aan volgende conventies:

- We starten een HTML-document altijd met `<!DOCTYPE html>`.
- We plaatsen de `<meta charset="utf-8">` altijd in het `<head>`-element.
- We hanteren een correcte volgorde en nesting van de elementen.
- We schrijven alle HTML-elementen en attributen in kleine letters.
- We zetten de waardes van de attributen altijd tussen dubbele aanhalingstekens.
- We sluiten de elementen met inhoud altijd via een eindtag.
- We plaatsen algemene teksten altijd in paragrafen.

Browsers zijn zeer vergevingsgezind. Hierdoor leveren kleine inbreuken niet onmiddellijk problemen op bij weergave. Toch zorgen we er best voor dat onze webpagina's voldoen aan de standaard, want enkel zo kunnen we een identieke weergave krijgen in alle browsers. Deze controle ten opzichte van de standaard noemen we 'validatie'.

Bij een validatie wordt een HTML5-document gecontroleerd op:

- Alle vereisten vastgelegd in de standaard
- Alle veelvoorkomende fouten
- Het bestaan van de gebruikte elementen en attributen

Om te valideren kunnen we gebruik maken van de ingebouwde validators in onze editor of van webbased validators. De bekendste webbased validator is <https://validator.w3.org/>. Via deze validator kunnen we zowel code als online en offline documenten controleren.

OPGELET! Een oefening is pas klaar als de code gevalideerd is zonder errors!

Structuur

Blokelementen en inline elementen

Er bestaan twee types van HTML-elementen:

- *Blokelementen*: deze elementen nemen altijd een nieuwe regel in beslag en maken gebruik van de volledige breedte van het scherm.

Dit is een blok element.

- *Inline elementen*: deze elementen worden in de huidige regel getoond en zijn even breed als de inhoud van het element.

Dit is een `inline element`.

Voorbeelden blokelementen

Paragraaf	<code><p>...</p></code>	Divisie	<code><div>...</div></code>
Kop 1	<code><h1>...</h1></code>	Blockquote/citaat	<code><blockquote>...</blockquote></code>
Kop 2	<code><h2>...</h2></code>	Preformatted	<code><pre>...</pre></code>
Kop 3	<code><h3>...</h3></code>	Ongeordende lijsten	<code></code>
Kop 4	<code><h4>...</h4></code>	Geordende lijsten	<code></code>
Kop 5	<code><h5>...</h5></code>		
Kop 6	<code><h6>...</h6></code>		

Voorbeelden inline elementen

Code blok	<code><code>...</code></code>	Afbeelding	<code></code>
Quote	<code><q>...</q></code>	Inputveld	<code><input /></code>
Hyperlink	<code><a>...</code>		<code><select></code>
Tekst-selectie	<code>...</code>	Line break	<code>
</code>
Urgent	<code>...</code>	Word break	<code><wbr /></code>
Superscript	<code><sup>...</sup></code>		
Subscript	<code><sub>...</sub></code>		
Uitgelicht	<code>...</code>		
Andere stemming	<code><i>...</i></code>		

Structuurelementen

In de vroegere HTML-versies werd er voor alles van structurele waarde het `<div>`-element gebruikt. Sinds HTML5 bestaan er specifieke structuurelementen.

- Hoofding `<header>...</header>`
- Navigatieblok `<nav>...</nav>`
- Hoofdgedeelte `<main>...</main>`
- Sectie (deel van een geheel) `<section>...</section>`
- Artikel (losstaand geheel) `<article>...</article>`
- Voetnoot of voettekst `<footer>...</footer>`
- Randinformatie `<aside>...</aside>`

Het is niet altijd eenvoudig om te bepalen wanneer we moeten kiezen voor welk structuurelement om een bepaalde inhoudsblok te definiëren. Vooral de keuze tussen sectie, artikel en randinformatie is voor interpretatie vatbaar. Daarom gebruiken we best de leidraad op de volgende pagina voor het kiezen van het juiste structuurelement.

Artikel	<p>De inhoud van dit element kan op zichzelf gelezen of gedeeld worden. De inhoud is onafhankelijk van de rest van de pagina.</p> <ul style="list-style-type: none"> • Bvb. Een recept in een kookboek.
Sectie	<p>De inhoud van dit element is een onderdeel van een geheel en is op zichzelf onvolledig.</p> <ul style="list-style-type: none"> • Bvb. Een hoofdstuk uit een boek.
Randinformatie	<p>De inhoud van dit element kan worden weggelaten. Het wordt gezien als iets 'extra', maar het is niet nodig om de inhoud te lezen of te begrijpen.</p> <ul style="list-style-type: none"> • Bvb. Een reclamebanner.
Hoofding	<p>De inhoud van dit element heeft een inleidend of verwijzend karakter.</p> <ul style="list-style-type: none"> • Bvb. De titel en subtitel van een blogpost.
Hoofdgedeelte	<p>De inhoud van dit element is dominant en essentieel.</p> <ul style="list-style-type: none"> • Bvb. De eigenlijke tekst van een blogpost.
Voettekst	<p>De inhoud van dit structuurelement omvat meta- of extra detailinformatie.</p> <ul style="list-style-type: none"> • Bvb. De copyrightinformatie onderaan een webpagina.
Navigatieblok	<p>De inhoud van dit element omvat navigatielinken.</p> <ul style="list-style-type: none"> • Bvb. De linken naar andere webpagina's binnen een website.

```

<body>
  <header>
    <h1>Paginatitel</h1>
    <p>Tekst</p>
  </header>
  <nav>
    <ul>
      <li><a href="link1.html">Link 1</a></li>
      <li><a href="link2.html">Link 2</a></li>
      <li><a href="link3.html">Link 3</a></li>
    </ul>
  </nav>
  <main>
    <section>

```

```

        <h2>Titel</h2>
        <p>Inhoud van de sectie</p>
    </section>
    <section>
        <h2>Titel</h2>
        <p>Inhoud van de sectie</p>
    </section>
</main>
<footer>
    <p>Paginavoetnoot</p>
</footer>
</body>

```

Hyperlinks

Via hyperlinks kunnen we navigeren naar andere webpagina's of bestanden. Om een hyperlink te bekomen, gebruiken we het `<a>`-element. Het meest gebruikte attribuut van het `<a>`-element is `href`, oftewel de 'hypertext reference'.

```

<a href="pagina1.html">Pagina 1</a>
<a href="http://www.andere-website.be/pagina2.html">Pagina 2</a>
<a href="/txt/tekst3.txt">Open tektbestand in map txt</a>

```

Een ander veelvoorkomend attribuut van het `<a>`-element is `target`. Dit attribuut geeft aan waar de link geopend wordt.

<code>_self</code>	In het huidige venster (default).
<code>_blank</code>	In een nieuw venster of tabblad van de browser.


```
<a href="http://www.px1.be" target="_blank">Website van de PXL</a>
```

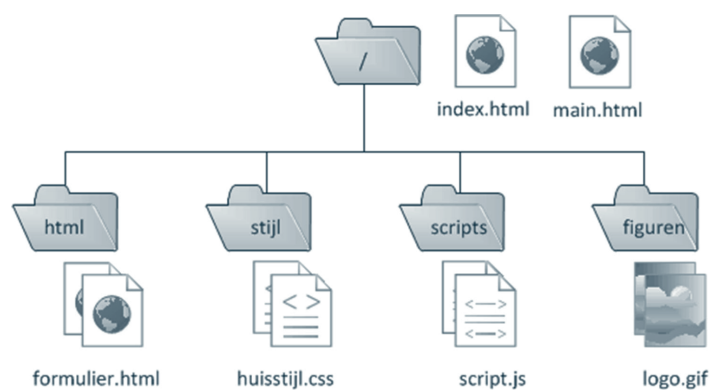
Absolute koppelingen

Een absolute koppeling verwijst naar een document op een webserver en gebruikt hiervoor een URL-adres.

```
<a href="http://www.andere-website.be/pagina2.html">Pagina 2</a>
```

Relatieve koppelingen

Een (document)relatieve koppeling verwijst naar een ander bestand, maar doet dit relatief ten opzichte van het bestand van waaruit de link wordt aangeklikt.



We maken een relatieve koppeling van 'index.html' naar 'main.html'. Het bestand waarnaar we linken staat in dezelfde map.

```
<a href="main.html">...</a>
```

We maken een relatieve koppeling van 'index.html' naar 'formulier.html'. Het bestand waarnaar we linken staat één map dieper in de folderhiërarchie (dalen).

```
<a href="html/formulier.html">...</a>
```

We maken een relatieve koppeling van 'formulier.html' naar 'index.html'. Het bestand waarnaar we linken staat één map hoger in de folderhiërarchie (stijgen).

```
<a href="../index.html">...</a>
```

Interne koppelingen

Een interne koppeling wordt gebruikt om naar een specifieke positie te gaan binnen eenzelfde of een ander document. Een interne koppeling bestaat altijd uit twee delen:

- Een *anker* die de positie binnen het document aangeeft.
- Een *hyperlink* die verwijst naar de positie binnen het document.

We maken een interne koppeling om te verwijzen naar een specifiek artikel op onze webpagina. Bij het klikken op de link, springt de webbrowser naar het artikel.

```
<h2>Inhoudsopgave</h2>
<ul>
  <li><a href="#deel1">deel 1</a></li>
  <li><a href="#deel2">deel 2</a></li>
</ul>
<article id="deel1">
  <h3>Deel 1: Inleiding</h3>
</article>
<article id="deel2">
  <h3>Deel 2: Structuur</h3>
</article>
```

We maken een interne koppeling om te verwijzen naar een specifiek artikel op een andere webpagina. Deze webpagina staat in dezelfde map als onze huidige webpagina. Het anker is reeds aanwezig op de andere webpagina.

```
<a href="index.html#deel2">deel 2</a>
```

We maken een interne koppeling om een top-link (knop op terug naar boven te gaan) toe te voegen aan een lange webpagina waarbij er een verticale scrollbar aanwezig is. Het plaatsen van een anker is niet nodig.

```
<a href="#">Terug naar boven</a>
```

Koppeling naar een e-mailadres

Bij een koppeling naar een e-mailadres voegen we een mailto-opdracht toe aan de link. De mailto-opdracht kunnen we naar wens uitbreiden met een 'cc', 'bcc' en 'subject'.

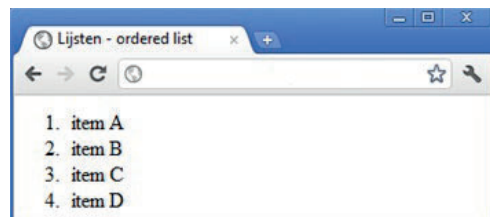
```
<a href="mailto:persoon1@bedrijf.be">...</a>  
  
<a href="mailto:persoon1@bedrijf.be?cc=persoon2@bedrijf.be&  
bcc=persoon3@bedrijf.be&subject=Nieuw%20bericht!">...</a>
```

Merk op dat de ampersand '&' geëncodeerd is als '&,' daar er geen speciale tekens kunnen gebruikt worden in HTML. Ook de spatie is geëncodeerd als '%20', omdat er geen spaties mogen voorkomen in een URL.

Lijsten

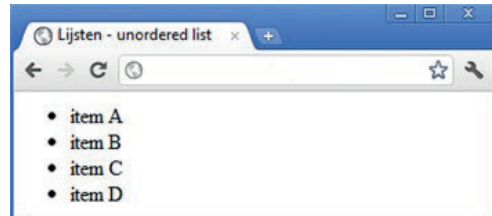
Geordende lijsten zijn lijsten met automatische nummering.

```
<ol>  
  <li>item A</li>  
  <li>item B</li>  
  <li>item C</li>  
  <li>item D</li>  
</ol>
```



Ongeordende lijsten zijn lijsten met opsommingstekens waarbij de volgorde van de items niet van belang is.

```
<ul>
  <li>item A</li>
  <li>item B</li>
  <li>item C</li>
  <li>item D</li>
</ul>
```



Definitielijsten zijn woordenlijsten met telkens een term gevolgd door de verklaring.

```
<dl>
  <dt>Term</dt>
  <dd>Definitie van woord</dd>
</dl>
```



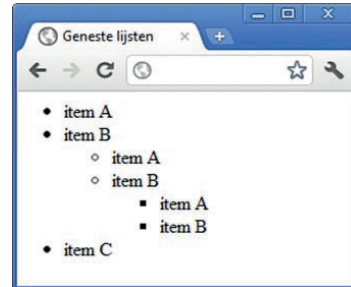
Geneste lijsten

Bij het nesten van een lijst binnen een andere lijst springen de verschillende niveaus automatisch in met telkens een ander opsomming (of nummering).

```

<ul>
  <li>item A</li>
  <li>item B
    <ul>
      <li>item A</li>
      <li>item B
        <ul>
          <li>item A</li>
          <li>item B</li>
        </ul>
      </li>
    </ul>
  </li>
  <li>item C</li>
</ul>

```



Afbeeldingen

We kunnen een afbeelding op onze webpagina plaatsen via de open tag ``. Hierbij zijn twee attributen altijd verplicht:

- *Het src-attribuut* geeft het pad naar de afbeelding die getoond moet worden.
- *Het alt-attribuut* geeft een alternatieve tekst die getoond moet worden als de afbeelding niet kan weergegeven worden.

```



```

Er kunnen verschillende afbeeldingsformaten op een webpagina geplaatst worden. Het gebruik van formaat is afhankelijk van de wensen en noden.

	GIF	JPEG	PNG
Aantal kleuren	256	16 miljoen	16 miljoen
Bestandsgrootte	Groter	Kleiner	Kleiner
Downloadtijd	Langer	Korter	Korter
Decompressietijd	Korter	Langer	Langer
Animatie	Aanbevolen	Afgeraden	Afgeraden
Transparantie	Gif89a	Niet mogelijk	Mogelijk
Logo's	Aanbevolen	Afgeraden	Afgeraden
Computertekening	Aanbevolen	Afgeraden	Afgeraden
Foto's	Afgeraden	Aanbevolen	Aanbevolen

Het figure-element

Het `<figure>`-element is een nieuw structuurelement in HTML5 en is bedoeld om foto's, illustraties, diagrammen en/of codelistings te groeperen.

```
<figure>
  
  
</figure>
```

Afbeelding als koppeling

Wanneer we een ``-element plaatsen binnen een `<a>`-element, werkt de afbeelding als een koppeling.

```
<a href="http://www.px1.be"></a>
```

Youtube-video's

Met een `<iframe>`-element kunnen we eenvoudig een Youtube-video invoegen op onze webpagina.

```
<iframe src="https://www.youtube.com/embed/VIDEO_ID" width="680"  
height="385"></iframe>
```

<https://www.youtube.com/watch?v=uyA01nH72NI>