

HTML (HyperText Markup Language)

De bouwstenen van elke webpagina is HTML. Dit is een taal die titel, van letter, van foto onderscheid, enzovoort. Ga maar eens naar een random (niet al te complexe) website zoals Wikipedia, vb.:

https://en.wikipedia.org/wiki/Rubber_duck_debugging . Door naar de 'broncode' van deze pagina te kijken, zien we waaruit deze bestaat. (Rechterklik > view pagesource/paginabron bekijken/weergeven of **ctrl+u** in de pagina). Hier is bijvoorbeeld een fragment van de html van de wiki pagina:

```
61 <li class="toclevel-1 tocsection-4"><a href="#External links"><span class="tocnumber">4</span> <span class="
62 </ul>
63 </div>
64 <p></p>
65 <h2><span class="mw-headline" id="In popular culture">In popular culture</span><span class="mw-editsectio
66 <table class="quote pullquote floatright" role="presentation" style="border-collapse: collapse; border-st
67 <tr style="text-align: left; vertical-align: top;">
68 <td style="color: #b2b7f2; font-size: 3.3em; font-family: 'Times New Roman', serif; font-weight: bold; padding
69 <td style="padding: 0 10px;"><i>Insights are often found by simply describing the problem aloud.</i></td>
70 <td style="color: #b2b7f2; font-size: 3.3em; font-family: 'Times New Roman', serif; font-weight: bold; padding
71 </tr>
72 <tr>
73 <td class="quotecite" colspan="3" style="padding-top: 10px; font-size: smaller; line-height: 1em; text-align
74 </tr>
75 </table>
76 <div class="thumb tleft">
77 <div class="thumbinner" style="width: 222px;"><a href="/wiki/File:StackExchange Rubber Duck Avatar April F
78 <div class="thumbcaption">
79 <div class="magnify"><a href="/wiki/File:StackExchange Rubber Duck Avatar April Fools 2018.png" class="in
80 The Stack Exchange rubber duck avatar featured on April 1, 2018</div>
81 </div>
82 </div>
83 <p>On April 1, 2018, <a href="/wiki/Stack Exchange" title="Stack Exchange">Stack Exchange</a> introduced
84 <div style="clear: both;"></div>
85 <h2><span class="mw-headline" id="See also">See also</span><span class="mw-editsection"><span class="mw-e
86 <div role="navigation" aria-label="Portals" class="noprint portal plainlist tright" style="margin: 0.5em 0
87 <ul style="display: table; box-sizing: border-box; padding: 0.1em; max-width: 175px; background: #f9f9f9; font-size
88 <li style="display: table-row"><span style="display: table-cell; padding: 0.2em; vertical-align: middle; text-align
89 </ul>
90 </div>
91 <ul>
```

Alle tekst tussen <> is html of css (style/stijl of opmaak).

Alle verschillende secties die door HTML gedefinieerd worden (**h1** -> **h5** voor titels, **a** voor links, ...) hebben een standaard opmaak (gedefinieerd door de browser). Maar omdat we niet willen dat elke pagina of site er hetzelfde uitziet kunnen we onze eigen 'opmaak' definiëren met css. Een pagina zonder css ziet er dan ook volledig anders uit. We nemen ons voorbeeld van hiervoor:

Dit is onze rubber duck debugging wikipedia pagina zonder css, enkel html.



Namelijk hier zichtbaar zien we een titel (**h1 = header 1** vermoedelijk), wat links naar andere pagina's (**a**), een foto (**img = image**), wat paragrafen tekst (**p = paragraph**), een ondertitel (h2), een lijst met enkele lijstelement zonder nummer (**ul = unnumbered list; li = list element**).

Verder willen we met html ook nog de grotere bouwstenen van een pagina definiëren, zoals bijvoorbeeld welke inhoud deel van de kop of header is, welke van de footer (onderaan de pagina) is, welke van de 'main' inhoud, welke deel van de navigatie van de site, enz. Hieronder gaan we kort even de opmaak van een basis html overlopen.

Elk html element zoals p kan ook ‘**attributen**’ bevatten die meer info over dat deel geven, bij sommige zijn deze verplicht zoals een **href** (referentie) bij een link (**a**) of foto (**img**), anders gebruiken we deze bij de css om blokken html van elkaar te onderscheiden. Als we bijvoorbeeld de eerste paragraaf blauw willen maken, maar de rest niet, geven we deze een **id** of **class** naam. `<p id="1eParagraaf">BLABLA</p>` ID's moeten uniek zijn in een html pagina, class namen mogen herhaald worden (als we bijvoorbeeld elke even paragraaf blauw willen, kunnen we al die een `class="evenP"` geven)

Een simpele html pagina

`<html lang="nl">` //Elke html pagina is omsloten door `<html></html>` om het te onderscheiden van niet html pagina's, hierin wordt ook vaak de taal van de pagina gedefinieerd (voor google en om bijv. blinden die software hebben om de pagina in de juiste taal te doen lezen)

`<head>` // Dit is nog voor de inhoud van de webpagina, hier definiëren we waarvan de html pagina gebruik van gaat maken, zoals bijvoorbeeld css pagina's

`<title></title>` // De titel van de html pagina (niet de h1 titel, degene van de tab in de browser)

`<link rel="stylesheet" href="css/reset.css">` // css pagina om de 'standaard' css af te zetten

`<link rel="stylesheet" href="css/style.css">` // onze eigen css of opmaak

`</head>`

`<body>` // Nu begint de html pagina inhoud

`<!-- Commentaar -->` // Om commentaar (niet deel van de site, wel broncode, tussen je code te zetten)

`<div>` // Om blokken code van elkaar te onderscheiden verdelen in 'generieke' delen/divisions

`<header>` // De 'kop' van de pagina, hierin komt meestal de naam site, logo, navigatie, ...

`<div>`

`<h1>` // de naam van de site: Oxfam in een h1

`OXFAM` // Dit is ook een link naar de home pagina

`</h1>`

`<nav id="siteNavigation">` // Navigatie, met een **id attribuut**

`` // unnumbered list, een lijst zonder nummers

`Onze Misse` //eerste lijstelement, een link in de navigatie die naar de html pagina onze_missie gaat

`Onze winkel`

``

`</nav>`

`</div>`

`</header>`

`</div>`

`<main id="index">` // de 'hoofd' inhoud van de pagina

`<div>`

`<article id="Missie">`

`<h2>Oxfam-Wereldwinkels</h2>`

`<p>Oxfam-Wereldwinkels wil rechtvaardige regels voor de internationale handel. Want eerlijke handel geeft producenten in het Zuiden sterke eigen benen en draagt op die manier bij tot duurzame ontwikkeling.</p>`

`<div>`

`<div>`

``

`<p>Miljoenen producenten in het Zuiden krijgen geen eerlijke kans op de wereldmarkt. Daardoor leven mensen er vaak in armoede.</p>`

`<p>Lees hier meer over de problemen.</p>`

`</div>`

`</article>`

`<article id="Producten">`

`<h2>Producten</h2>`

`<div>`

`<p>Producten van eerlijke handel zijn onze specialiteit. Nergens anders vind je er zoveel bij elkaar. Met het brede assortiment van ons huismerk Oxfam Fair Trade en andere fairtrademarken tonen we dat eerlijke handel mogelijk is.</p>`

`</div>`

`<div class="bestelKnop">`

`<h3>Bestel nu hier je favoriete producten</h3>`

`</div>`

`</article>`

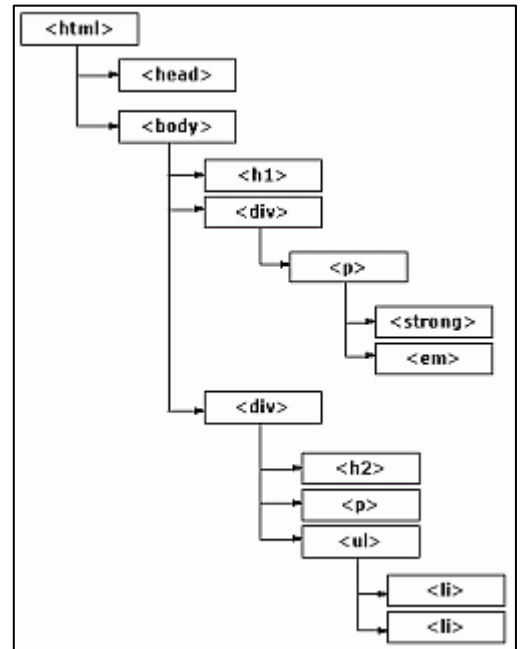
`<aside>` // soms willen we een kolom naast de hoofd inhoud, met extra inhoud

```

<article id="Recepten">
  <h3><a href="recepten.html">Recepten</a></h3>
  <div>
    <p></p>
  </div>
</article>
</aside>
</main>

<footer id="floatingFooter" class="grad"> // de footer (onderaan de pagina)
<section id="Contact">
  <h4>Contact</h4>
  <address>
    Naamsesteenweg 133, <br>
    3001 Heverlee
  </address>
  <p>016 29 41 02</p>
</section>
<section id="Openingsuren">
  <h4>Openingsuren</h4>
  <div>
    <table> // tabel
    <tr>
      <th>Vrijdag</th>
      <td>10:00-18:00</td>
    </tr>
    </table>
  </div>
</section>
<div id="BottomFooter"> Copyright Maria Verdonck, UCLL, 2016 </div>
</footer>
</body>
</html>

```



HTML 'Boom'

TL;DR : HTML 'tags' zijn de bouwstenen van een site. Tags definiëren wat de inhoud van die tag is, zoals bijvoorbeeld titels **h1-6**; bevatten meta informatie zoals een url van een link **link** of id's/klassen, ...; geven de pagina een standaard opmaak die we met **css** kunnen aanpassen.

CSS (Cascading Style Sheet)

Nu we een onze inhoud mooi in secties verdeeld hebben met HTML tags, kunnen we elke sectie zijn eigen opmaak gaan geven. Onder opmaak verstaan we niet alleen kleur, font, achtergrond, etc. , maar ook positie en soms ook bepaald gedrag (bijvoorbeeld een link die van kleur verandert als je erover zweeft of erop geklikt hebt of hoe dingen van plaats veranderen als we onze browser kleiner maken).

Css kan rechtstreeks in de html tags gedefinieerd worden (zie de eerste screenshot), maar meestal doen we dit in een apart css bestand, waarin we in de head naar verwijzen, zodat we een uniforme opmaak over de hele site hebben.

Om te definiëren welk deel van de site we stijl aan het geven zijn gebruiken we css **selectors**, die zijn ofwel de **html tags** zelf, ofwel de **#id** en **.class attributen**. Of een combinatie hiervan. Css pagina's bestaan dan uit dit soort elementen:

```

Selector {
  Property: value;
  Property2: value2;
}

```

Css is **cascading** omdat css eigenschappen worden overgenomen / overgeërfd. Als we bijvoorbeeld zeggen dat alle tekst in main rood zal zijn, zal alle tekst in de paragrafen die in main zitten, rood zijn. **Tenzij** we **daarna** in de css definiëren dat deze bijvoorbeeld blauw moeten zijn. Css eigenschappen kunnen ook 'overschreven' worden. Als we meer naar beneden in de css dan weer zeggen dat tekst in p groen moet zijn, zullen deze groen zijn, ook al hadden we ze als specifiek bij naam als blauw gedefinieerd.

HTML heeft dus een bepaalde hiërarchie (te zien aan insprongen van onze html voorbeeld hierboven) die bepaald hoe dingen worden 'overgeërfd' van 'ouder' op 'kind'. (zie html 'boom' hierboven)

Een fragment van een css pagina

```
p {  
    Color: black; // Een kleine selectie kleuren kan je bij naam definiëren, de rest met rgb(a), hexa #, ...  
    Size: 12px; // Formaten van tekst kunnen in pixels, cm, percentages, ...  
}  
/*floatingFooter*/ // Commentaar in een css pagina is tussen /* Commentaar */  
#floatingFooter { // #ID die in html gedefinieerd werd  
    display: flex;  
    justify-content: space-around; // De inhoud wordt evenredig verdeeld over de breedte van het scherm  
    padding-bottom: 30px; // padding is ruimte tussen de tekst en de rand van de html blok  
    border: 2px dotted red; // de rand van het html blok zal rood, gestippeld en rood zijn  
    border-radius: 40px 40px 0 0; // de rand zal een ronding in linker -en rechterbovenhoek hebben, niet onderaan  
}  
#floatingFooter h4 { // de hoofding h4 in de html blok met id floatingFooter  
    padding-bottom: 5px;  
    text-decoration: underline; // tekst wordt onderlijnd  
}  
#floatingFooter {  
    color: rgb(37, 34, 34); //red, green, blue percentages geven een kleur weer  
}
```

TL;DR Om te definiëren welk deel van de site we stijl aan het geven zijn gebruiken we css **selectors** (html tags, id/class attributen of combinatie hiervan). **Selector { Property: value; }**

ZIE OOK!

ZIE OOK CSS+HTML Cheat Sheet voor een overzicht van HTML tags en basis CSS selectors/properties

Voor een uitgebreide uitleg over alles HTML, CSS (en meer): <https://www.w3schools.com/html/default.asp>