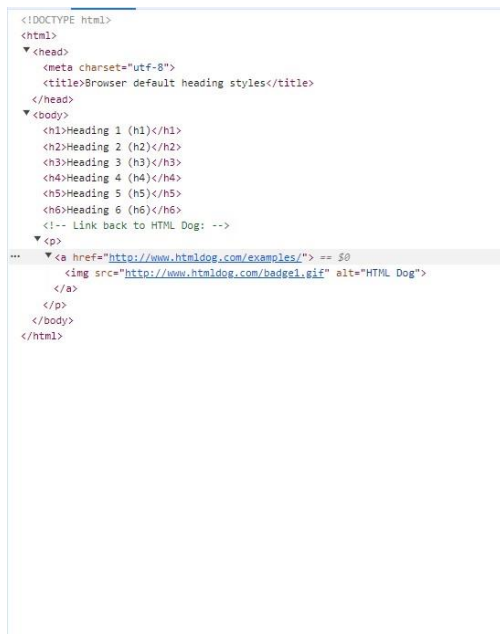


2.3 Opdracht 1

Surf naar <https://www.HTMLdog.com/examples/headings1.HTML> en bekijk de broncode. Vergelijk deze met de response body uit de originele HTTP-request voor deze pagina op het “Netwerk” tabblad van de Chrome Developer tools (het netwerk tabblad en HTTP requests kwamen in het vorige labo aan bod).



Ze zijn hetzelfde.

2.4 Opdracht 2

Maak een nieuw project aan in Webstorm en neem de basisstructuur van een HTML5 webpagina over in een nieuw HTML bestand. Bekijk dit vervolgens ook in Chrome.

```
1  <!DOCTYPE html>
2  <html lang="nl">
3  <head>
4      <meta charset="UTF-8" />
5      <title>het werkt</title>
6  </head>
7  <body>
8      gevonden
9  </body>
10 </html>
```

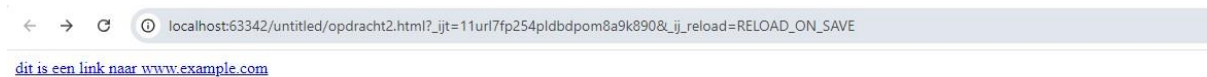


2.6 Opdracht 3

Plaats telkens een van de bovenstaande drie voorbeeldelementen in het body element van ons basisdocument, bewaar en bekijk telkens de pagina in Chrome.

Denk eraan op refresh te klikken, de browser weet immers niet dat het document op schijf is veranderd.

De browser negeert trouwens extra whitespace (lege ruimte zoals spaties of tab en newline karakters), zowel tussen woorden op eenzelfde regel als over meerdere regels. Het heeft dus geen zin om te proberen de layout te beïnvloeden met extra spaties of extra lege regels in het HTML bestand. Dit lijkt een nadeel maar is voor ons al programmeurs een voordeel omdat het ons toelaat de HTML code te schikken op een manier die ze voor ons duidelijk(er) maakt in een editor, zonder dat dit gevolgen heeft voor de gebruiker die deze in de browser bekijkt.



Je kan een HTML bestand dat op je harde schijf staat trouwens op twee manieren bekijken in je browser:

- via het file:/// protocol o door erop te dubbelklikken in de Windows verkenner

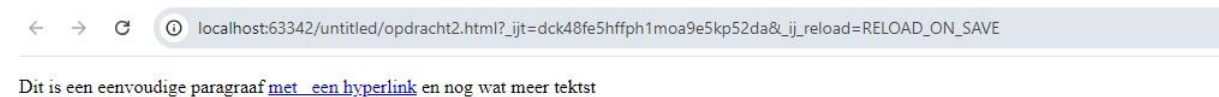
- via het http:// protocol op de host 'localhost' op poort 63342 o door in Webstorm 'Open in browser' te kiezen

Wat is het verschil tussen beide? Via de localhost ga je via een lokale webserver.

2.7 Opdracht 4

Plaats in een HTML document wat extra spaties tussen 2 woorden in een paragraaf en ga na dat dit daadwerkelijk geen invloed heeft op de pagina voorstelling in de browser. Plaats daarna enkele lege regels tussen 2 woorden en vergewis je ervan dat ook dit geen effect heeft op de visualisatie.

Je kan trouwens extra whitespace forceren door ` ` en `
` te gebruiken voor resp. Extra spaties en nieuwe regels.



Gebruik deze echter niet als gemakzuchtige layout trucs om de ruimte tussen pagina onderdelen te corrigeren; daarvoor gebruiken we immers CSS!

2.8 Opdracht 5

Maak een HTML-document waarin de volgende tekst staat:

Gebruik ` ` en `
` echter niet voor layout doeleinden, daarvoor gebruiken we immers CSS!

Je zult hiervoor enkele entity references uit Sectie 3.1 moeten gebruiken.

← → ↻ ⓘ localhost:63342/untitled/opdracht2.html?_ijt=dck48fe5hffph1moa9e5kp52da&_ij_reload=RELOAD_ON_SAVE

Gebruik ` ` en `
` echter niet voor layout doeleinden, daarvoor gebruiken we immers CSS!

```
1  <!DOCTYPE html>
2  <html lang="nl">
3  <head>
4      <meta charset="UTF-8" />
5      <title>het werkt</title>
6  </head>
7  <body>
8      <p>Gebruik &nbsp; en <br /> echter niet voor layout doeleinden, daarvoor gebruiken we immers CSS! </p>
9  </body>
10 </html>
```

2.10 Opdracht 6

Waarom zijn de volgende uitdrukkingen niet correct genest?

- $(4 \times 2 + 2($ *er zijn geen haakjes dicht voor de geopende haakjes*
- $(4 \times 2(+ 2$ *er zijn geen haakjes dicht voor de geopende haakjes*
- $(4 \times (2 + 2)$ *er is geen haakje dicht voor het eerst geopende haakje*
- $(4 \times (2 + 2($ *er zijn geen haakjes dicht voor de eerste twee geopende haakjes*

Waar we in de wiskunde meestal maar 1 soort tag gebruiken (namelijk ronde haakjes) zijn er in de HTML taal verschillende soorten begin- en eindtags mogelijk: elk soort element heeft een eigen begin- en eindtag!

2.11 Opdracht 7

Veronderstel dat we in een tekst verschillende soorten haakjes mogen mengen: `()`, `{}` en `[]`. De voorwaarde is echter dat ze correct genest moeten zijn. Zijn de haakjes in de volgende teksten correct genest?

- `{[]}[]` ja
- `{{{()}}` nee
- `{{()}}` ja

- `(){}[]` nee
- `{{([])}` nee

2.12 Opdracht 8

Veronderstel een HTML document met daarin, o.a.,

```
<span>hier komt een</span><a href="">link</a><span> te staan</span>
```

(Terzijde: is een spatie net voor het woord "te" eigenlijk nodig? Probeer eens uit!)

Als je dit niet doet plakken de woorden "link" en "te" aan elkaar

De auteur beslist plots dat de tekst "een link" benadrukt moet worden dmv. een `` element en verandert bovenstaande naar:

```
<span>hier komt <strong>een </strong><a href="">link</a></strong><span> te staan</span>
```

Waarom is dit niet correct, en hoe moet het dan wel? Dit is niet correct omdat je geen `` eindtag hebt voor "een" en geen `` begintag hebt voor `link`. Het zou moeten zijn `hier komt een link te staan`

2.13 Opdracht 9

Reproduceer de volgende pagina (incl. afbeelding):

<http://www.HTMLdog.com/examples/headings1.HTML> en gebruik hierbij onze HTML5 basis structuur. Bewaar dit als "voorbeeld headings.HTML" en open dit in de browser. Ziet het er correct uit? *Ja*



Plaats de afbeelding in een folder 'images' in je project. Ziet het er nog steeds correct uit?
Nee

Heb je de waarde van het src attribuut van het element moeten aanpassen? Ja

```
1  <!DOCTYPE html>
2  <html lang="nl">
3  <head>
4      <meta charset="UTF-8" />
5      <title>Browser default heading styles</title>
6  </head>
7  <body>
8      <h1>Heading 1 (h1)</h1>
9      <h2>Heading 2 (h2)</h2>
10     <h3>Heading 3 (h2)</h3>
11     <h4>Heading 4 (h4)</h4>
12     <h5>Heading 5 (h5)</h5>
13     <h6>Heading 6 (h6)</h6>
14     <p>
15         <a href="http://www.htmldog.com/examples/">
16             
17         </a>
18     </p>
19 </body>
20 </html>
```

2.14 Opdracht 10

Wat is er verkeerd in onderstaand HTML fragment? Je maakt een genummerde lijst in een ongenummerde lijst terwijl de genummerde lijst eigenlijk deel moet zijn van de ongeordende lijst. Je moet de hele genummerde lijst in tags zetten.

een

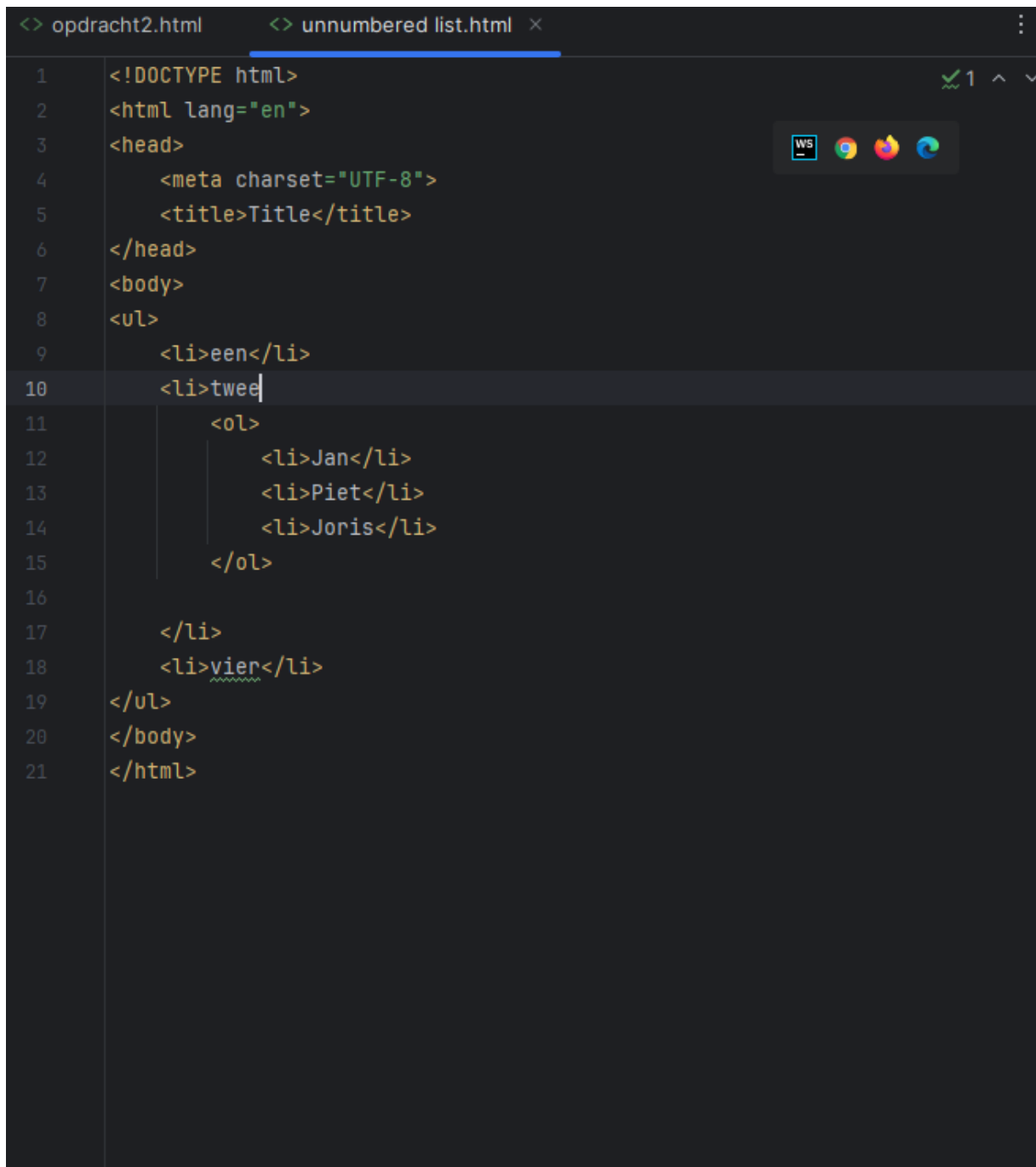
twee

jan

piet

joris

vier



```
<> opdracht2.html  <> unnumbered list.html  x
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6  </head>
7  <body>
8  <ul>
9      <li>een</li>
10     <li>twee
11         <ol>
12             <li>Jan</li>
13             <li>Piet</li>
14             <li>Joris</li>
15         </ol>
16     </li>
17     <li>vier</li>
18 </ul>
19 </body>
20 </html>
```

2.18 Opdracht 11

We zullen eens kijken hoe de website van Microsoft informatie bijhoudt in de browser. Surf naar:

<https://www.microsoft.com>

Open nu de Chrome Developer Tools en kijk op het Application tabblad bij 'local storage' en 'cookies' welke data deze website door de browser laat bewaren. Je zult zien dat er per domein waar de webpagina gebruik van maakt, aparte data wordt bijgehouden. Wijzig eens

de regio/taal van de pagina (linksonderaan de webpagina van microsoft) en kijk opnieuw naar de cookies. Zie je waar de gekozen taalinstelling wordt bijgehouden? *Bij de cookies.*

Opdracht 12

Surf naar: <https://www.lg.com>. Ga naar de televisie sectie. Daar kun je een overzicht bekomen van al hun actuele toestellen die je ook kan vergelijken. Je zult zien dat je verschillende modellen kunt vergelijken door ze aan te vinken en dan op de Compare knop te klikken. Merk op dat je gerust modellen op verschillende pagina's kunt aanvinken om toe te voegen aan je vergelijking.

Ergens moet deze webapplicatie dus bijhouden (over meerdere pagina's heen) wat we aangevinkt hebben. Open de Chrome Developer Tools en kijk op het Application tabblad bij 'local storage' en 'cookies' welke data daar bewaard wordt. Vink een aantal toestellen aan en kijk wat er verandert. Om de veranderingen te zien zul je wellicht op die 'local storage' en 'cookies' tabbladen op de kleine refresh knop moeten klikken (dus niet de grote refresh knop naast je adresbalk, die laadt immers de ganse pagina opnieuw in!).

Waar precies bewaart de webapplicatie welke toestellen je aanvinkte ter vergelijking? Ben je nog andere herkenbare informatie tegengekomen op die beide tabbladen? Zoja, welke? *Dit wordt bewaard bij de local storage.*

2.19 Opdracht 13

We zullen eens kijken hoe men bij Amazon informatie bijhoudt over je winkelkarretje.

Surf naar <https://www.amazon.com>.

Mocht je daar een account hebben en ingelogd zijn, gelieve uit te loggen voor deze opdracht. Plaats een product in je winkelwagen en kijk opnieuw bij local storage en cookies. Wordt de inhoud van je winkelwagen daar ergens bewaard? Zoniet, waar dan wel denk je? *Dit wordt waarschijnlijk bewaard in hun server.*

Als je het tabblad sluit en op een nieuw tabblad weer naar Amazon.com surft, zul je zien dat je product nog steeds in het winkelkarretje zit. Hoe weten ze bij Amazon eigenlijk welke winkelwagen bij jou hoort? Je bent immers niet ingelogd op hun site! *Ze linken je winkelkarretje aan je ip adres.*