

Basis statische website

Realiseren

hoofdstuk

5

Lijsten, links en navigatie





Algemene informatie

Onderwerp	Links
Leerdoel(en)	<ol style="list-style-type: none">1. De student kan in eigen woorden uitleggen waarom links gebruikt worden op websites.2. De student kan links coderen die verwijzen naar andere websites, andere webpagina's binnen dezelfde website en naar specifieke plaatsen op een webpagina zelf door gebruik te maken van ankers.3. De student kan links opmaken door middel van CSS, waarbij onder andere de hover en visited pseudoselectoren worden toegepast.4. De student kan een webpagina inladen op een andere webpagina door middel van een iframe.
Vereiste voorkennis	De student moet een basiskennis hebben over HTML en CSS waarbij de student in staat is om eenvoudige teksten te kunnen coderen en opmaken. Deze kennis is opgedaan in hoofdstuk 2 en 3.
Kwalificatiedossier	<ul style="list-style-type: none"><input type="checkbox"/> B1-K1-W1: Plant werkzaamheden en bewaakt de voortgang<input type="checkbox"/> B1-K1-W2: Ontwerpt software<input checked="" type="checkbox"/> B1-K1-W3: Realiseert (onderdelen van) software<input type="checkbox"/> B1-K1-W4: Test software<input type="checkbox"/> B1-K1-W5: Doet verbetervoorstellen voor de software <input type="checkbox"/> B1-K2-W1: Voert overleg<input type="checkbox"/> B1-K2-W2: Presenteert het opgeleverde werk<input type="checkbox"/> B1-K2-W3: Reflecteert op het werk



Inhoudsopgave

Algemene informatie	2
Inhoudsopgave	3
Introductie	4
Inhoud	4
Soorten lijsten	4
Geordende lijsten	4
Ongeordende lijsten	5
Lijsten nesten	6
Linken naar andere websites	9
Links openen in een nieuw venster	9
Linken naar webpagina's binnen dezelfde website	9
Linken naar specifieke plaatsen op een webpagina	11
Links opmaken	13
Webpagina inladen op een andere webpagina	14
Navigatie	16
Verticaal menu maken	17
Horizontaal menu maken	19
Opmaak van de navigatie	20



Introductie

Een website bestaat eigenlijk altijd uit meerdere webpagina's. In de vorige hoofdstukken heb je al geleerd om eenvoudige webpagina's op te zetten die voorzien van een structuur, opmaak en interactie. Deze webpagina's moeten uiteraard ook aan elkaar gekoppeld. In dit hoofdstuk leer je verschillende soorten links maken zodat een bezoeker gemakkelijk door een website kan navigeren. In thema 2 ga je dit verder uitwerken in de vorm van een menu.

Inhoud

Soorten lijsten

Lijsten kunnen gebruikt worden om tekst beter te structureren. Er wordt onderscheid gemaakt tussen geordende lijsten en ongeordende lijsten. Lijsten herken je aan opsommingstekens, de tekens die voor ieder onderdeel van de lijst staan.

Geordende lijsten

Geordende lijsten zijn lijsten waarbij elk item in de lijst genummerd is. De lijst kan bijvoorbeeld bestaan uit een aantal stappen voor een recept die in de juiste volgorde moeten worden uitgevoerd. Geordende lijsten kunnen er bijvoorbeeld zo uit zien:

- | | | | |
|-----------|-----------|-----------|-------------|
| 1. Stap 1 | a. Stap 1 | A. Stap 1 | I. Stap 1 |
| 2. Stap 2 | b. Stap 2 | B. Stap 2 | II. Stap 2 |
| 3. Stap 3 | c. Stap 3 | C. Stap 3 | III. Stap 3 |

De geordende lijst wordt gemaakt met het ol-element (ordered list).
Een item (de stappen zelf) worden gemaakt met het li-element (list item).

```
<h1>
  Geordende lijst
</h1>
<ol>
  <li>
    Stap 1
  </li>
  <li>
    Stap 2
  </li>
  <li>
    Stap 3
  </li>
</ol>
```



Resultaat:

Geordende lijst

1. Stap 1
2. Stap 2
3. Stap 3

De opsommingstekens kunnen in CSS worden aangepast. Dat komt verderop aan bod in deze reader. Standaard wordt er voor een geordende lijst gekozen voor getallen.

Ongeordende lijsten

Ongeordende lijsten zijn lijsten die beginnen met een opsommingsteken (in plaats van letters of cijfers die een volgorde aangeven). Er wordt dus geen specifieke volgorde aangegeven.

Ongeordende lijsten kunnen er bijvoorbeeld zo uit zien:

- | | | | |
|--------|--------|--------|--------|
| • Item | ○ Item | ▪ Item | ✚ Item |
| • Item | ○ Item | ▪ Item | ✚ Item |
| • Item | ○ Item | ▪ Item | ✚ Item |

De ongeordende lijst wordt gemaakt met het ul-element (unordered list). Ook hier worden de items met het li-element gemaakt.

```
<h1>  
  Ongeordende lijst  
</h1>  
<ul>  
  <li>  
    Item  
  </li>  
  <li>  
    Item  
  </li>  
  <li>  
    Item  
  </li>  
</ul>
```

Resultaat:

Ongeordende lijst

- Item
- Item
- Item

Ook hier kunnen de opsommingstekens in CSS worden aangepast. Standaard wordt er voor een ongeordende lijst gekozen voor rondjes.



Lijsten nesten

Je kunt lijsten ook in elkaar nesten (een lijst in een lijst plaatsen). Hiermee kun je bijvoorbeeld nog een onderverdeling maken in een recept.

1. Snijd onderstaande groente in kleine blokjes
 - Ui
 - Wortel
 - Paprika
2. Doe de groente in de wok
3. Bak de groente gedurende 8 minuten

```
<h1>
  Geneste lijst
</h1>
<ol>
  <li>
    Snijd onderstaande groente in kleine blokjes
    <ul>
      <li>
        Ui
      </li>
      <li>
        Wortel
      </li>
      <li>
        Paprika
      </li>
    </ul>
  </li>
  <li>
    Doe de groente in de wok
  </li>
  <li>
    Bak de groente gedurende 8 minuten
  </li>
</ol>
```

Let er dus op dat je een nieuwe lijst aanmaakt binnen een li-element.

Resultaat:

Geneste lijst

1. Snijd onderstaande groente in kleine blokjes
 - Ui
 - Wortel
 - Paprika
2. Doe de groente in de wok
3. Bak de groente gedurende 8 minuten



Opsommingsteken veranderen (standaard)

Zoals je gezien hebt wordt er automatisch al een opsommingsteken gebruikt bij lijsten. Dit kun je in CSS aanpassen met het attribuut **list-style-type**. Er wordt hierbij geen onderscheid gemaakt tussen geordende lijsten en ongeordende lijsten. Om je code logisch te houden ga je bij een ongeordende lijst uiteindelijk geen cijfers of letters gebruiken als opsommingsteken. Daarom maak je over het algemeen gebruik van onderstaande attribuutwaarden. Als selector kies je de lijst zelf.

Geordende lijsten

```
ol  
{  
  list-style-type: disc;  
}
```

Geordende lijst (ol)	
Voorbeeld	Attribuutwaarde voor list-style-type
	none
1 2	decimal
01 02	decimal-leading-zero
a b	lower-alpha
A B	upper-alpha
i. ii.	lower-roman
I II	upper-roman

Ongeordende lijsten

```
ul  
{  
  list-style-type: decimal;  
}
```

Ongeordende lijst (ul)	
Voorbeeld	Attribuutwaarde voor list-style-type
	none
•	disc
○	circle
■	square



Opsommingsteken veranderen (eigen afbeelding)

Je kunt ook een eigen gekozen afbeelding gebruiken als opsommingsteken. Dit doe je alleen op ongeordende elementen (ul). Dit doe je in CSS met het attribuut **list-style-image**.

```
ul
{
  list-style-image: url("../images/kw1cIcon.png");
}
```

Lijst met afbeelding als opsommingsteken

- ▲▲▲ Item 1
- ▲▲▲ Item 2
- ▲▲▲ Item 3

Let op dat je de afmeting van het plaatje dat je wilt gebruiken als opsommingsteken (ongeveer) gelijk is aan de lettergrootte. Anders zal het opsommingsteken niet mooi voor de items staan.

Positioneren van het opsommingsteken

Daarnaast kun je het opsommingsteken op twee verschillende manieren positioneren. Dit doe je in CSS met het attribuut **list-style-position**. Als attribuutwaarde geef je **outside** (standaard) of **inside** op.

Positioneren van het opsommingsteken	
Outside	Inside
1. Dit opsommingsteken is naast het item geplaatst. Daardoor loopt deze tekst niet onder het opsommingsteken door.	1. Dit opsommingsteken is in het item geplaatst. Daardoor loopt deze tekst onder het opsommingsteken door.



Linken naar andere websites

Links worden met het a-element gemaakt. De bezoeker kan op alles klikken wat in het a-element staat. Met het attribuut href geef je aan welke pagina je wilt linken.

URL van de webpagina waar de link naar verwijst.

Tekst die zichtbaar is in de browser waar de bezoeker op kan klikken.

```
<a href="http://www.kw1c.nl">KW1C website</a>
```

Zorg ervoor dat de linktekst goed omschrijft waar je de bezoeker naar toe stuurt. Een link met de tekst "klik hier" is dan ook niet aan te raden.

Geef daarnaast ook altijd de volledige url op inclusief "http://" of "https://". We noemen dit absolute url's. HTTP is het standaard protocol van een website om data door te sturen. Daarom begint bijna iedere url van een webpagina met http. HTTPS is een andere, beveiligde, manier om informatie te versturen. De S staat voor Secured. Informatie wordt bij dit protocol versleuteld zodat alleen de verstuurder en ontvanger de informatie kunnen lezen (en geen derde partijen die kwaad willen).

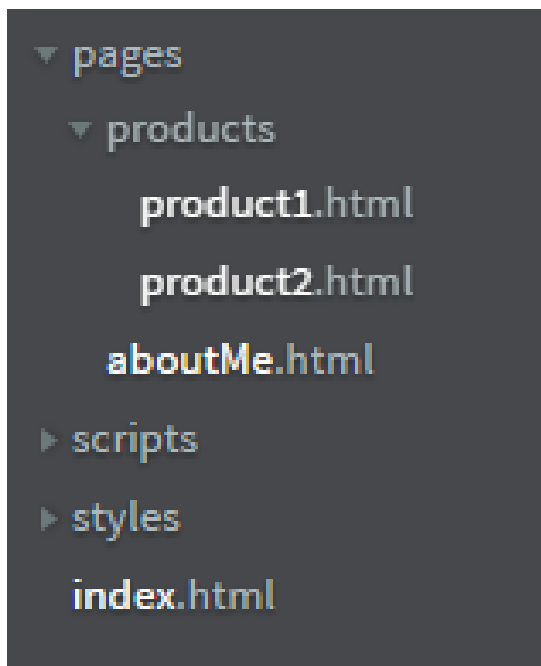
Links openen in een nieuw venster

Als je wilt dat een link in een nieuw venster opent, dan kun je het attribuut target in de openingstag van het a-element gebruiken. De waarde van dit attribuut moet dan _blank zijn. Probeer dit echter zoveel mogelijk te voorkomen. Het wordt eigenlijk alleen gebruikt als je bezoekers naar een compleet andere website stuurt. Het is voor de bezoeker dan gemakkelijk om weer terug te keren naar de oorspronkelijke website.

```
<a href="http://www.kw1c.nl" target="_blank">KW1C website</a>
```

Linken naar webpagina's binnen dezelfde website

Wanneer je een link naar andere pagina's binnen dezelfde website maakt, hoef je de domeinnaam (bijvoorbeeld <http://www.kw1c.nl>) niet in de URL op te geven. Je kunt gebruikmaken van een verkorte schrijfwijze die bekend staat als een relatieve URL. Als alle pagina's van de site in dezelfde map zouden staan, dan is de waarde van het attribuut href gewoon de naam van het bestand (inclusief .html). Echter vinden wij het belangrijk dat je alle webpagina's behalve de index.html in een map genaamd pages plaatst. Dit om de bestanden netjes te structureren. Je geeft dan op waar de pagina zich ten opzichte van de huidige pagina bevindt.



Hieronder volgt een aantal voorbeelden van links die je kunt maken op basis van de structuur die in bovenstaande afbeelding is gebruikt.

Link van index.html naar aboutMe.html

Om van index.html naar aboutMe.html te gaan dien je eerst naar de map pages te gaan.

```
<a href="pages/aboutMe.html">Linktekst</a>
```

Link van index.html naar product1.html

Om van index.html naar product1.html te gaan dien je eerst naar de map pages te gaan en daarna naar de map products.

```
<a href="pages/products/product1.html">Linktekst</a>
```

Link van aboutMe.html naar product1.html

Om van aboutMe.html naar product1.html te gaan dien je eerst naar de map products te gaan.

```
<a href="products/product1.html">Linktekst</a>
```

Link van product1.html naar product2.html

Omdat de bestanden zich allebei in dezelfde map bevinden hoeft je alleen de bestandsnaam op te geven.

```
<a href="product2.html">Linktekst</a>
```



Link van product1.html naar aboutMe.html

Om van product 1.html naar aboutMe.html te gaan dien je een map terug te gaan. Dat doe je door in de url ../ toe te passen

```
<a href="../aboutMe.html">Linktekst</a>
```

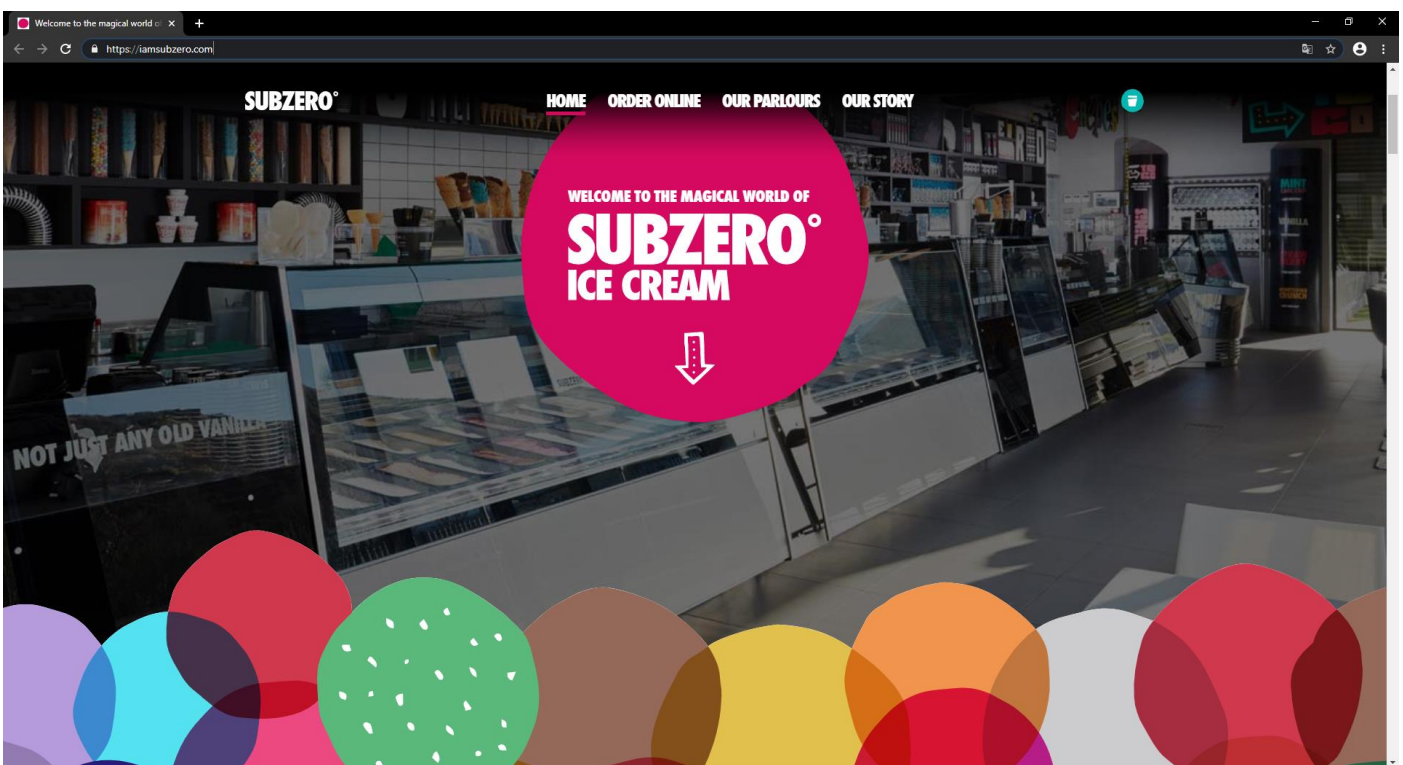
Link van product1.html naar index.html

Om van product 1.html naar index.html te gaan dien je twee mappen terug te gaan. Je dient nu dus 2 keer ../ toe te passen.

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

Linken naar specifieke plaatsen op een webpagina

Je kunt op een webpagina de gebruiker ook naar een specifieke plaats op die webpagina laten navigeren. Dit zie je vaak gebeuren bij zogeheten one-page websites. Dit zijn websites waarbij de webpagina's als het ware onder elkaar geplakt zijn, in plaats van daadwerkelijk andere webpagina's. Op moment van schrijven is de website van Subzero Ice Cream een mooi voorbeeld van een one-page website. Als je in het menu klikt zie je dat de browser naar beneden scrolt totdat het juiste anker is bereikt.



<https://iamsubzero.com/>



Als je links naar ankers opgeeft die zich op dezelfde pagina bevinden dan kun je als link direct het anker opgeven. Dat doe je met een hashtag gevolgd door het id van het element waar je naar toe wilt scrollen.

```
<a href="#bottom">Ga naar het einde</a>
<a href="#intro">Ga naar de intro</a>
<a href="#core">Ga naar de kern</a>
<p id="intro">
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed
  pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim.
  Fusce est. Vivamus a tellus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis
  egestas. Proin pharetra nonummy pede. Mauris et orci. Aenean nec lorem.
  In porttitor. Donec laoreet nonummy augue. Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc.
  Mauris eget neque at sem venenatis eleifend. Ut nonummy. Fusce aliquet pede non pede. Suspendisse dapibus lorem
  pellentesque magna. Integer nulla. Donec blandit feugiat ligula. Donec hendrerit, felis et imperdiet euismod, purus ipsum
  pretium metus, in lacinia nulla nisl eget sapien.
  Donec ut est in lectus consequat consequat. Etiam eget dui. Aliquam erat volutpat. Sed at lorem in nunc porta tristique.
  Proin nec augue. Quisque aliquam tempor magna. Pellentesque habitant morbi tristique senectus et netus et malesuada fames
  ac turpis egestas. Nunc ac magna. Maecenas odio dolor, vulputate vel, auctor ac, accumsan id, felis. Pellentesque cursus
  sagittis felis.
<p id="core">
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed
  pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim.
  Fusce est. Vivamus a tellus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis
  egestas. Proin pharetra nonummy pede. Mauris et orci. Aenean nec lorem.
  In porttitor. Donec laoreet nonummy augue. Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc.
  Mauris eget neque at sem venenatis eleifend. Ut nonummy. Fusce aliquet pede non pede. Suspendisse dapibus lorem
  pellentesque magna. Integer nulla. Donec blandit feugiat ligula. Donec hendrerit, felis et imperdiet euismod, purus ipsum
  pretium metus, in lacinia nulla nisl eget sapien.
  Donec ut est in lectus consequat consequat. Etiam eget dui. Aliquam erat volutpat. Sed at lorem in nunc porta tristique.
  Proin nec augue. Quisque aliquam tempor magna. Pellentesque habitant morbi tristique senectus et netus et malesuada fames
  ac turpis egestas. Nunc ac magna. Maecenas odio dolor, vulputate vel, auctor ac, accumsan id, felis. Pellentesque cursus
  sagittis felis.
<p id="end">
  The end!
</p>
<a id="bottom" href="#top">Ga naar boven</a>
```

Je ziet dat onderaan de webpagina een #top anker wordt opgegeven terwijl er geen element bestaat met een id top. Toch scrollt de website weer naar boven als je op de linktekst klikt. Dat komt omdat alle moderne browsers weten dat ontwikkelaars hiermee het begin van een webpagina bedoelen.

Je kunt ook navigeren naar ankers van een andere webpagina. In dat geval voeg je achter de url het anker toe.

```
<a href="pages/products/product1.html#reviews">Bekijk de reviews</a>
```

Je navigeert nu naar de pagina product.html en vervolgens scrollt de browser direct door naar het anker reviews (het element met een id reviews).

Vloeiende scrollbeweging

Bij het navigeren naar een anker zie je dat de browser direct, zonder vloeiende scrollbeweging, naar het element navigeert. Veel ontwikkelaars kiezen ervoor om wel zo'n vloeiende scrollbeweging in te stellen. Je kunt dat doen door gebruik te maken van onderstaande CSS-code. LET OP: dit werkt alleen in moderne browsers!

```
html
{
  scroll-behavior: smooth;
}
```



Links opmaken

Uiteraard kun je links ook opmaken met CSS. Echter kent een link bepaalde toestanden. Hiermee bedoelen we bijvoorbeeld onbezocht, bezocht en aanwijzen. Door middel van zogeheten pseudoclass-selectoren kun je links opmaken in een bepaalde toestand. **Wegens veiligheidsoverwegingen staan browsers niet alle opmaak toe bij deze selector. Het kan dus zijn dat bepaalde opmaak niet werkt!**

Onbezochte links

Met de pseudoclass-selector `a:link` kun je links opmaken die nog niet bezocht zijn door de gebruiker.

```
a:link
{
    color: blueviolet;
}
```

Je denkt nu misschien dat dit hetzelfde is als onderstaande code.

```
a
{
    color: blueviolet;
}
```

Echter selecteer met de selector `a` alle soorten links, dus ook links die bijvoorbeeld al bezocht zijn. Als je de `a`-selector gebruikt in combinatie met onderstaande pseudoclass-selectoren zorg er dan voor dat deze altijd onder de `a`-selector geplaatst worden. Op deze manier zorg je ervoor dat de code van de `a`-selector overschreven wordt.

Bezochte links

Met de pseudoclass-selector `a:visited` kun je links opmaken die nog niet bezocht zijn door de gebruiker.

```
a:visited
{
    color: darkgreen;
}
```

Aanwijzende links

Hiermee bedoelen we de toestand dat de gebruiker zijn muis op een link houdt. De style hiervan kun je opmaken met de pseudoclass-selector `a:hover`.

```
a:hover
{
    color: orangered;
}
```

Onderstreping



Zoals je in hoofdstuk 3 al geleerd heb kun je met het CSS-attribuut `text-decoration` ervoor zorgen dat je een tekst onderstreept. Met hetzelfde attribuut kun je er ook voor zorgen dat je de onderstreping van een link juist weghaalt.

```
a:hover
{
  color: orangered;
  text-decoration: none;
}
```

Webpagina inladen op een andere webpagina

Met een link stuur je de gebruiker naar een andere locatie van je eigen website, of zelfs een compleet andere website. Je kunt ook een andere webpagina inladen op een webpagina. Een veelvoorkomend voorbeeld is bijvoorbeeld een contactpagina waar vervolgens Google Maps wordt ingeladen. Je laadt dan als het ware een andere webpagina in een webpagina. Dit wordt vaak gedaan met het `iframe`-element. Met het attribuut `src` kun je aangeven welke webpagina je wilt inladen. Met het `seamless`-attribuut (werkt alleen in HTML 5) zorg je ervoor dat scrolbalken niet zichtbaar zijn. Dit is een bijzonder attribuut omdat je hier geen attribuutwaarde bij opgeeft (zie onderstaande voorbeeld). Uiteraard dien je de `iframe` ook een hoogte en breedte te geven. Dit doe je uiteraard in CSS. Je kunt daarvoor de CSS-attributen `height` en `width` gebruiken.

HTML:

```
<body>
  <h1>
    Contact
  </h1>
  <iframe src="https://www.google.com/maps/embed?
pb=!1m18!1m12!1m3!1d275.43511148552466!2d5.286525237176266!3d51.690532281738506!2m3!1f0!2f
0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x0%3A0xc69337c87886a627!2sKW1C!5e0!3m2!1snl!2snl!
4v1601834523032!5m2!1snl!2snl" seamless>
  </iframe>
</body>
```

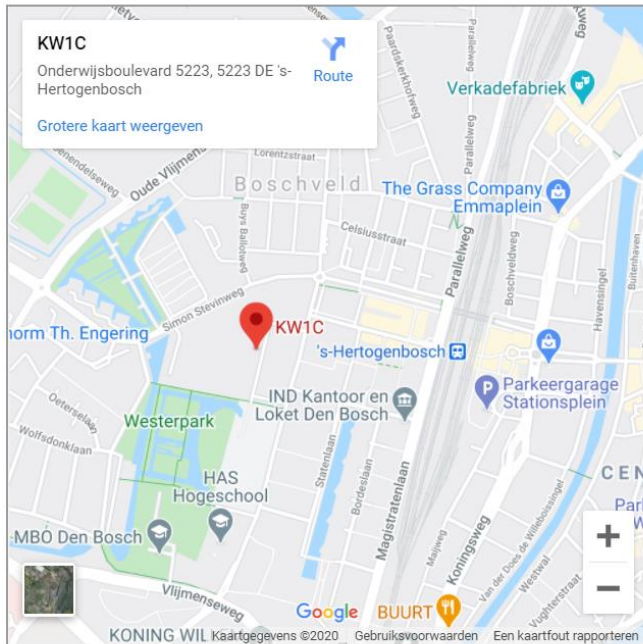
CSS:

```
iframe
{
  width: 500px;
  height: 500px;
}
```

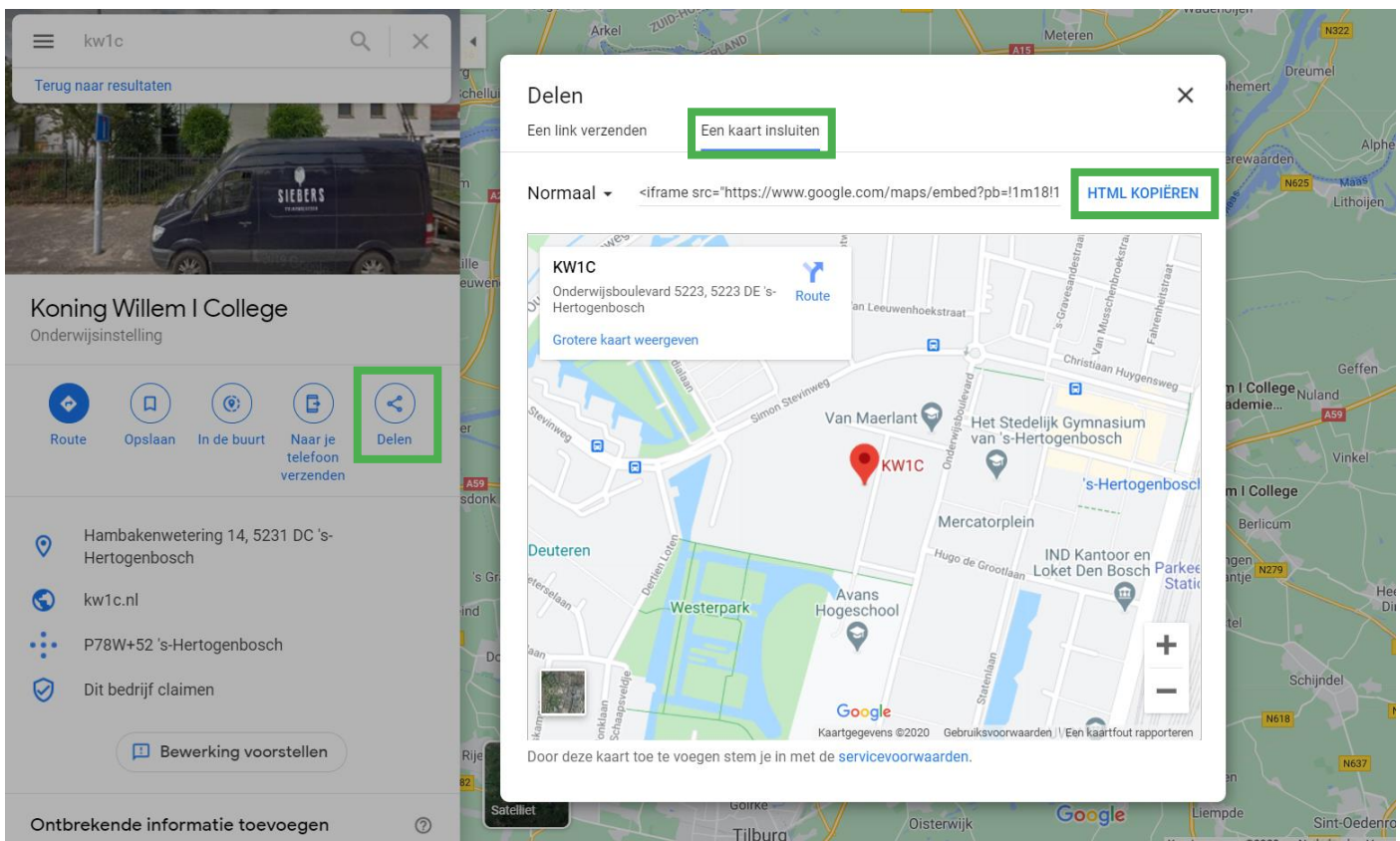



Resultaat:

Contact



Tip: Je kunt via Google Maps het iframe element ook laten maken.



Je ziet dat er ook andere attributen in het iframe-element worden opgenomen. Onderzoek of deze attributen bijdragen aan het resultaat dat je graag wilt.



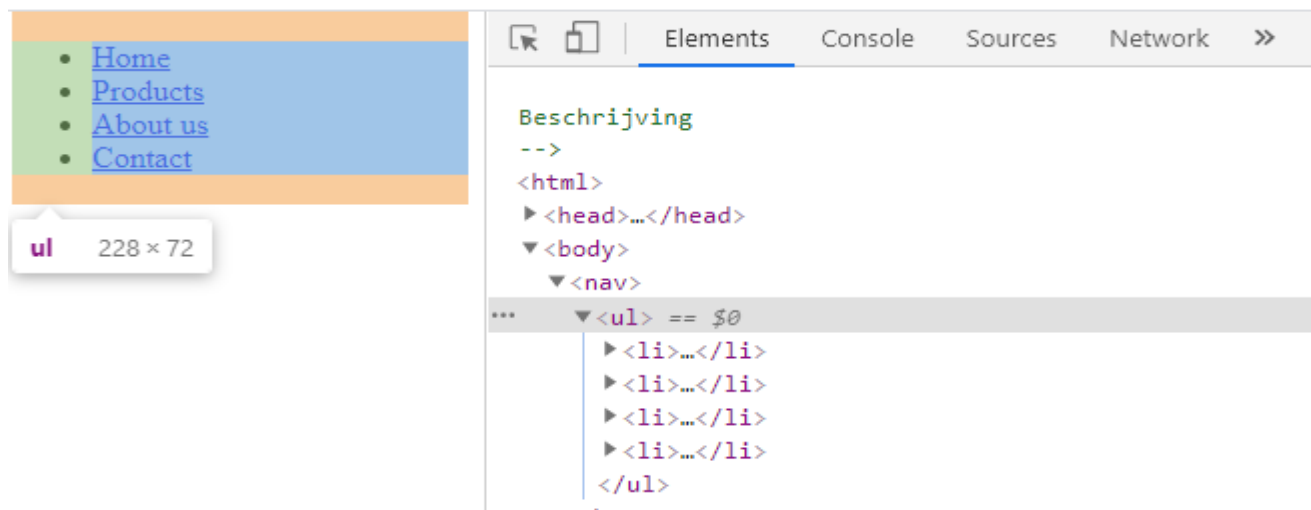
Navigatie

Voor het maken van een menu maak je gebruik van een ul-element met daarin li-elementen. Deze li-elementen worden straks je menu-items die de gebruiker moet linken naar een andere pagina. Daarom plaats je in de li-elementen een a-element. Plaats de lijst vervolgens in een nav-element om aan te geven dat het onderdeel is van je navigatie.

```
<nav>
  <ul>
    <li>
      <a href="pages/home.html">
        Home
      </a>
    </li>
    <li>
      <a href="pages/products.html">
        Products
      </a>
    </li>
    <li>
      <a href="pages/aboutUs.html">
        About us
      </a>
    </li>
    <li>
      <a href="pages/contact.html">
        Contact
      </a>
    </li>
  </ul>
</nav>
```

Daarna ga je de lijst opmaken in CSS. Het is je misschien al opgevallen maar standaard staat een lijst niet helemaal tegen de zijkant van zijn ouder (in dit geval de body).

Door gebruik te maken van de developer tools kun je zien dat er een groen/gele balk aan de zijkant van het element staat en onder en boven een oranje balk. De groen/gele ruimte noemen we een padding, de oranje ruimte een margin. Daarover leer je in het volgende hoofdstuk meer.

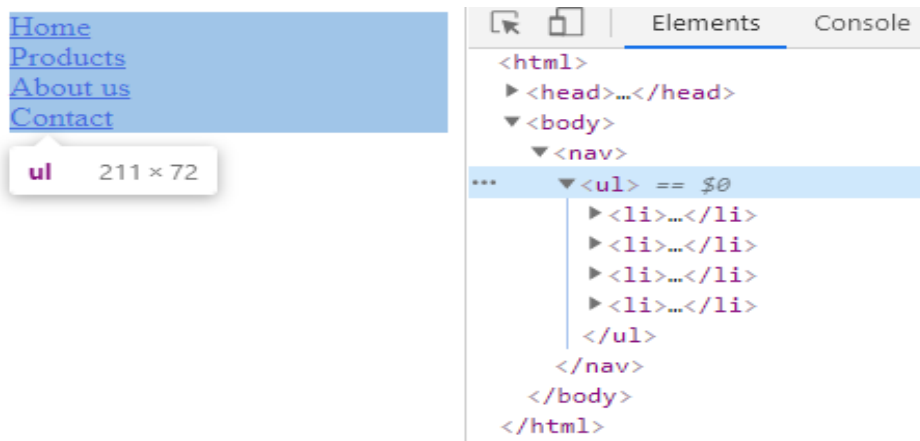




Het is wel belangrijk om die ruimte nu te verwijderen. Dat doe je door onderstaande CSS-code te gebruiken.

```
ul
{
    padding: 0px;
    margin: 0px;
}
```

Als je nu nogmaals in de developer tools kijkt zie je dat de ruimte verdwenen is.



Je ziet nu dat de opsommingstekens ook verdwenen zijn. Dat komt omdat de opsommingstekens geplaatst zijn in de ruimte van de padding. Die heb je nu verwijderd, waardoor ook de opsommingstekens verdwijnen. Als je een menu maakt wil je ook dat de opsommingstekens verdwijnen, dus dat is nu ook opgelost!

Vanuit deze basis wordt nu uitgelegd hoe je hier een verticaal menu en een horizontaal menu van kunt maken.

Verticaal menu maken

Het lijkt al enigszins op een verticaal menu. Het is wel aan te raden om het menu minder breed en aantrekkelijker te maken. Omdat het ul-element (en daardoor ook het nav-element) de gehele breedte van de pagina heeft, kun je er in principe niet meer gemakkelijk andere elementen naast gaan plaatsen.

Door het nav-element een breedte te gaan geven kun je ervoor zorgen dat je er wel elementen naast kunt gaan zetten (hoe je dat doet leer je volgend hoofdstuk). Daarnaast kun je het menu natuurlijk ook een achtergrondkleur geven.

```
nav
{
    width: 200px;
    background-color: rgb(77,182,172);
}
```

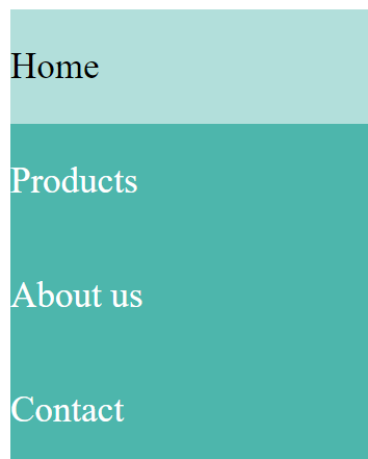


Door de elementen verder op te maken kun je er uiteindelijk een mooi verticaal menu van maken.

```
nav
{
    width: 200px;
    background-color: rgb(77,182,172);
}
ul
{
    padding: 0px;
    margin: 0px;
}
a
{
    color: white;
    display: block;
    font-size: 20px;
    text-decoration: none;

    /* Extra: Als je de "knoppen" wat groter wilt maken, maar de tekst toch verticaal
    uitgelijnd wilt hebben kun je gebruik maken van padding. */
    padding-top: 20px;
    padding-bottom: 20px;
}
a:hover
{
    color: black;
    background-color: rgb(178,223,219);
}
```

Resultaat:





Horizontaal menu maken

In deze paragraaf wordt onderstaande code gebruikt als basiscode. Dit is dezelfde code als bovenstaande. De html structuur van de navigatie zal dus voor iedere navigatie hetzelfde zijn.

```
<nav>
  <ul>
    <li>
      <a href="pages/home.html">
        Home
      </a>
    </li>
    <li>
      <a href="pages/products.html">
        Products
      </a>
    </li>
    <li>
      <a href="pages/aboutUs.html">
        About us
      </a>
    </li>
    <li>
      <a href="pages/contact.html">
        Contact
      </a>
    </li>
  </ul>
</nav>
```

```
ul
{
  padding: 0px;
  margin: 0px;
}
```

Om de items uit de lijst naast elkaar te krijgen kun je verschillende technieken gebruiken. Één daarvan is het gebruik van de css eigenschap **display**. Standaard zal een li-element de hele breedte innemen en het volgende li element op een nieuwe regel plaatsen. Door dit te veranderen naar inline zal het element alleen zijn eigen breedte innemen en het volgende element ook niet meer naar een nieuwe regel verplaatsen. Hierdoor zullen de menuitems dus naast elkaar komen te staan.

```
li
{
  display: inline;
}
```

Een andere techniek die je regelmatig tegen zal komen is het gebruik van **float**.

```
li
{
  float: left;
}
```

Resultaat:

- [Home](#)[Products](#)[About us](#)[Contact](#)

Door het gebruik van het float attribuut gaat de li zich ook gedragen als een soort inline-element alleen behoudt hij wel de stijl van de lijst. Echter zal float ook wat gevolgen hebben voor de elementen die na de navigatie in de code staan. Om die reden is het nodig om bij het ul element de waarde van het attribuut **overflow** op **hidden** in te stellen.



Opmaak van de navigatie

Om vervolgens ruimte tussen de items te creëren kun je de items bijvoorbeeld een vaste breedte mee geven.

```
li
{
    width: 25%;
    float: left;
}
```

In dit geval wordt er gekozen voor 25% omdat er 4 menu-items zijn (samen 100%).

Resultaat:

- [Home](#)
- [Products](#)
- [About us](#)
- [Contact](#)

Bij het gebruik van float zie je dat er nu weer opsommingstekens worden weergegeven. Dit kun je weghalen door op het **ul** of **li** element de waarde van het attribuut **list-style-type** of **list-style** op **none** te zetten.

```
li
{
    width: 25%;
    float: left;
    list-style-type: none;
}
```

Nu kun je nog extra opmaak toevoegen om het menu mooier te maken.

```
nav
{
    background-color: rgb(77,182,172);
}
ul
{
    padding: 0px;
    margin: 0px;
    list-style: none;
    overflow: hidden;
}
li
{
    float: left;
    width: 25%;
}
a
{
    color: white;
    display: block;
    font-size: 20px;
    text-decoration: none;
    text-align: center;

    /* Extra: Als je de "knoppen" wat groter wilt maken, maar de tekst toch verticaal
    uitgelijnd wilt hebben kun je gebruik maken van padding. */
    padding-top: 20px;
    padding-bottom: 20px;
}
a:hover
{
    color: black;
    background-color: rgb(178,223,219);
}
```

Resultaat:

Home	Products	About us	Contact
------	----------	----------	---------