

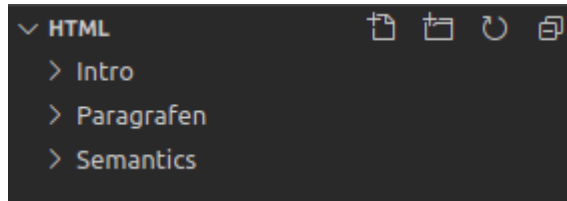
# HTML



## Handout Cursist

# HTML handout

In deze handout vind je alle HTML opdrachten terug die je moet maken. Het is de bedoeling dat je bij het beantwoorden van de vragen zo uitgebreid mogelijk te werk gaat en altijd ervoor zorgt dat je goed research hebt gedaan. Om geordend te werk te gaan willen we graag dat je werkt met een goed opgebouwde mappenstructuur. Dus elk onderwerp in een eigen bijbehorende map. Je mappenstructuur zal er als volgt eruit te komen zien:



## Inhoud

### 1. Intro HTML

Leren over de basis termen, concepten, semantics en opbouw van HTML.

### 2. HTML Elementen

Leren hoe HTML elementen en attributen werken.

### 3. Forms & Tabellen

Leren hoe forms en tabellen gebruikt worden in HTML

### 7. Eindopdracht

De eindopdracht bestaat uit het uittekenen en nabouwen van volledige webpagina's en het navigeren naar andere HTML pagina's.

## Opdrachten

### Onderdeel 1:

Er zijn regels hoe HTML toegepast wordt. De opdrachten zijn gemaakt om meer inzicht te krijgen hoe deze regels zijn opgesteld. Maak de opdrachten hieronder.

Om mee te beginnen, lees [hier](#) wat HTML inhoudt.

### Intro

1. Wat is het verschil tussen het internet en het world wide web?
2. Wat is het verschil tussen HTML, CSS en JavaScript?
3. Wat is een HTML tag en een HTML element? Hoe open je een HTML tag en hoe sluit je een HTML tag. Maak hiervoor voorbeelden en leg uit.
4. HTML gebruikt basis tags om structuur op te bouwen. De `<html>`, `<head>` en `<body>`. Geef voor elke element aan waarvoor het dient.
5. Op HTML elementen kun je attributen gebruiken. Wat zijn HTML attributen ? Geef minimaal 5 attributen en maak voorbeelden waar je de attributen in toepast.
6. Waarom is een goede HTML Structuur belangrijk? Is [deze code](#) een goede structuur? Leg uit.
7. Wat is het verschil tussen inline en block elementen?
8. Wat betekent nesting? Codeer een voorbeeld met nesting erin.

### Semantics

9. Sinds de komst van HTML5 zijn er elementen die betekenissen hebben. Wat wordt er bedoeld met HTML semantics? Waarom is het gebruik van semantics belangrijk?

Lees meer over [semantics markup](#)

10. Wat voor content zou je in de volgende elementen gebruiken?

Geef in je antwoord voor elk element een toelichting.

- `<header>`
- `<footer>`
- `<main>`
- `<nav>`
- `<section>`

- <aside>
- <article>

## Headers

Het gebruik van headers is belangrijk voor zoekmachines en de gebruikers van je website.

11. Onderzoek en leg uit waarom headers zowel belangrijk zijn voor zoekmachines als voor de gebruikers.

Voor het aanmaken van een text-header, in het Nederlands een subkop of kop, kan je gebruik maken van een Header tag. Je kunt verschillende grootte koppen creëren.

12. Onderzoek hoe je een text-header aanmaakt en reproduceer het volgende resultaat:

# Headings opdrachten

## Heading 1

### Heading 2

#### Heading 3

##### Heading 4

###### Heading 5

###### Heading 6

## Paragrafen

13. Onderzoek wat een paragraaf tag is en hoe hij gebruikt wordt.  
Geef hier een voorbeeld bij.
14. Onderzoek en leg het verschil tussen het stoppen van alle tekst in één paragraaf tag, en het opsplitsen van de tekst in meerdere tags uit.

## 15. Reproduceer het volgende resultaat door gebruik te maken van text-headers en de paragraaf tags.

### Mijn eerste website

#### Mijn eerste sub-kop

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras sed pulvinar nisl, volutpat vulputate lectus. Morbi eget dapibus lectus. Fusce neque nunc, tempor et sapien eu, faucibus placerat nibh. Mauris nec sem mattis, congue libero eu, tincidunt nunc. Suspendisse potenti. Sed eget pellentesque sapien. Maecenas rutrum, augue eget dapibus feugiat, mauris arcu volutpat felis, vel tincidunt ipsum orci ut erat. Nulla auctor enim semper congue vestibulum.

#### Mijn tweede sub-kop

Nam hendrerit augue sagittis, elementum risus eu, pretium risus. Cras malesuada quam at sapien pellentesque elementum. Aliquam ipsum erat, faucibus et leo id, dapibus ultrices metus. Nunc porta magna non imperdiet tristique. Nullam congue ornare ante. In convallis felis non magna semper pharetra. Suspendisse dapibus elementum lacus, in tempor leo lobortis in. Proin semper risus tristique, tempus odio sit amet, dignissim magna. In non mi vehicula, auctor mi vel, lacina lectus.

Suspendisse potenti. Nam sit amet magna leo. Pellentesque dignissim, metus id pulvinar malesuada, sapien magna vestibulum erat, non elementum lacus ligula ac est. In a feugiat tellus, in interdum mauris. Fusce pharetra eros quis dapibus fringilla. Integer a neque vitae sapien porttitor lobortis vitae ac arcu. Sed eu egestas augue, quis pellentesque dui. Praesent eu diam eu risus fringilla cursus. Donec a orci vitae nibh ullamcorper tristique. Nunc auctor mi ut elit rutrum placerat. Cras at venenatis nibh.

## Horizontal rule

We willen nu een footer toevoegen onderaan onze pagina. Onderzoek hoe een horizontal rule werkt en start aan de onderstaande opdracht.

## 16. Voeg een Horizontal rule toe en reproduceer het volgende resultaat.

### Mijn eerste website

#### Mijn eerste sub-kop

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras sed pulvinar nisl, volutpat vulputate lectus. Morbi eget dapibus lectus. Fusce neque nunc, tempor et sapien eu, faucibus placerat nibh. Mauris nec sem mattis, congue libero eu, tincidunt nunc. Suspendisse potenti. Sed eget pellentesque sapien. Maecenas rutrum, augue eget dapibus feugiat, mauris arcu volutpat felis, vel tincidunt ipsum orci ut erat. Nulla auctor enim semper congue vestibulum.

#### Mijn tweede sub-kop

Nam hendrerit augue sagittis, elementum risus eu, pretium risus. Cras malesuada quam at sapien pellentesque elementum. Aliquam ipsum erat, faucibus et leo id, dapibus ultrices metus. Nunc porta magna non imperdiet tristique. Nullam congue ornare ante. In convallis felis non magna semper pharetra. Suspendisse dapibus elementum lacus, in tempor leo lobortis in. Proin semper risus tristique, tempus odio sit amet, dignissim magna. In non mi vehicula, auctor mi vel, lacina lectus.

Suspendisse potenti. Nam sit amet magna leo. Pellentesque dignissim, metus id pulvinar malesuada, sapien magna vestibulum erat, non elementum lacus ligula ac est. In a feugiat tellus, in interdum mauris. Fusce pharetra eros quis dapibus fringilla. Integer a neque vitae sapien porttitor lobortis vitae ac arcu. Sed eu egestas augue, quis pellentesque dui. Praesent eu diam eu risus fringilla cursus. Donec a orci vitae nibh ullamcorper tristique. Nunc auctor mi ut elit rutrum placerat. Cras at venenatis nibh.

#### Footer

Deze website is gebouwd vanuit de TechGrounds Academy

## Link tags

In onze footer willen we graag linken naar de TechGrounds website. Onderzoek hoe een link tag werkt.

## 17. Voeg een link tag toe en reproduceer het volgende resultaat in de footer.

### Footer

Deze website is gebouwd vanuit de [TechGrounds Academy](#)

Zodra een gebruiker op de link klikt wordt de pagina weergegeven in de huidige browser window. Door gebruik te maken van attributen kan de link geopend worden in bijvoorbeeld een nieuw tabblad. Voorbeelden van HTML attributen zijn bijvoorbeeld een image tag die een “alt” attribuut heeft.

## 18. Onderzoek welke 4 attributen een link tag kan hebben, leg uit wat deze doen en voeg deze als voorbeeld toe op je website.

## Onderdeel 2 - HTML Elementen

### Recap dag 1

Gisteren hebben we kennis gemaakt met de basics van HTML en enkele HTML elementen. We zullen nog de belangrijkste onderdelen herhalen en de gemaakte opdrachten bespreken. We zullen nog de onderwerpen links, afbeeldingen, video's en audio los behandelen.

### Links

HTML links zijn hyperlinks. Dit betekent dat de gebruiker door middel van een klik van het ene document naar een ander document toe kan springen. Standaard verandert de muispijl ook naar een muishand. Een link hoeft niet altijd tekst te zijn, het kan ook een afbeelding zijn.

```
<a href="www.techgrounds.nl">link text</a>
```

De belangrijkste attribuut van een a element is de href attribuut. Dit definieert de bestemming van de link. De tekst tussen de <a> </a> tags is het enige onderdeel dat zichtbaar is voor de gebruiker

Een veelgebruikte attribuut voor de link tag is de target attribuut. Met de target attribuut kun je specificeren op welke manieren de webbrowser de link zal openen. Als je deze attribuut niet definieert zal de browser de webpagina openen in de huidige browser window. Als je dit wilt wijzigen kun je de target attribuut aanpassen. Hieronder enkele voorbeelden met hoe je dit doet en de resultaten hiervan:

```
<a href="www.techgrounds.nl" target="_self">link text</a>
```

Opent het document in de huidige browser. Niet definiëren en op deze manier definiëren bereik je dus hetzelfde mee.

```
<a href="www.techgrounds.nl" target="_blank">link text</a>
```

Opent het document in een nieuwe window of tabblad.

```
<a href="www.techgrounds.nl" target="_parent">link text</a>
```

Opent het document in het parent frame. Het parent frame is de frame dat om het huidige frame heen zit.

```
<a href="www.techgrounds.nl" target="_top">link text</a>
```

Top zorgt ervoor de de webpagina zal worden ingeladen in de body van je huidige pagina.

Je kan ook naar een onderdeel van je webpagina linken en door middel van een klik op de link automatisch springen naar een ander onderdeel van je webpagina. Dit is hoe navigatiebalken werken en dit onderdeel zal verder toegelicht worden in het onderdeel lijsten en navigatiebalken.

## Media

Online kan Multimedia teruggevonden in verschillende vormen. Multimedia op een website is geluid, muziek, video's, films, animaties en nog veel meer. Voor elk multimedia formaat zijn er bestandsoorten die ondersteund worden door de webbrowser. Enkele veel voorkomende zijn: MP3, WebM, Wav en MP4.

### <video>

De <video> tag wordt gebruikt om een video weer te geven op een website. De video tag is een container tag waarin je een <source> element definieert waar je het video formaat definieert. Als je tekst toevoegt tussen de video tag zal die tekst weergegeven worden als de browser het video element niet ondersteunt.

*Attributen:*

### autoplay

Wordt gebruikt om de video automatisch te starten

### width & height

Wordt gebruikt om de width en height vast te stellen. Als dit niet wordt vastgesteld kan de webpagina na het inladen van de video de content omlaag duwen.

### controls

Wordt gebruikt in de video tag om een afspeel, pauzeer en volume knop toe te voegen.

### <source>

`src="movie.mp4"` - Definieert welke videobestand je wilt gebruiken,

`type="video/mp4"` - Definieert het soort type bestand.

### <audio>

Wordt gebruikt om audio af te spelen op een website door middel van de source tag. Heeft dezelfde attributen als video behalve width & height.

### <iframe>

De eenvoudigste manier om video's af te spelen in HTML is om Youtube te gebruiken. Hier kunnen we een iframe voor gebruiken. Een iframe, een inline frame, is een HTML document of webpagina dat in een ander HTML document geplaatst is. Je kan dit zien als een kleine doos waarin we een andere pagina kunnen weergeven. Voorbeelden hiervan zijn Youtube video's op andere websites en advertenties. Hier worden in werkelijkheid een onderdeel van een andere webpagina weergegeven op jouw website.

### title

Wordt gebruikt voor screenreaders om te zien wat de content van een iframe is.

### width & height

Wordt gebruikt om de width en height vast te stellen van de iframe. Standaard worden pixels gebruikt.

### style

Wordt gebruikt om de styling aan te passen. Zo kan je de height en width hier exact aanpassen, border verwijderen, positioneren en meer.

### name

Zie onderstaand voorbeeld. Hier hebben we in de link tag als target de naam van ons iframe meegegeven. Dit zorgt ervoor dat de iframe gewijzigd wordt naar de link zijn href zodra hierop geklikt wordt. De link wordt dus geopend in onze iframe.

```
<iframe src="content.htm" name="iframe_a" title="Iframe Example"></iframe>
```

```
<p><a href="www.techgrounds.nl" target="iframe_a">Techgrounds.nl</a></p>
```

### Youtube

Om een Youtube video weer te geven moet je naar de video navigeren die je wilt weergeven, op delen klikken en dan op embed klikken. Youtube genereert automatisch de iframe code die hiervoor nodig is. De attributen kan je opzoeken en wijzigen tot je het resultaat bereikt hebt dat je wilt bereiken.

### Section

Een <section> representeert een alleenstaande sectie binnen een HTML document. Vaak heeft een sectie zijn eigen heading. Door het gebruik maken van secties in je HTML document kan je je webpagina opdelen in verschillende onderdelen. Hieronder volgt een voorbeeld:



```
<section>
<h2>WWF History</h2>
<p>The World Wide Fund for Nature (WWF) is an international organization working on
issues regarding the conservation, research and restoration of the environment, formerly
named the World Wildlife Fund. WWF was founded in 1961.</p>
</section>
```

```
<section>
<h2>WWF's Symbol</h2>
<p>The Panda has become the symbol of WWF. The well-known panda logo of WWF
originated from a panda named Chi Chi that was transferred from the Beijing Zoo to the
London Zoo in the same year of the establishment of WWF.</p>
</section>
```

Een sectie kan gezien worden als het opdelen van een webpagina en doen we per nieuw onderwerp of onderdeel. Het opdelen van een webpagina in onderdelen in secties geeft dus aan dat er een nieuw onderdeel begint.

## Div

Door middel van het gebruiken van een div kan een website ook opgedeeld worden. Een div verschilt van een section in dat deze opdeling geen semantic betekenis hoeft te hebben. De opdeling wordt vaak opgesteld om onderdelen van de website te stylen of te manipuleren via JavaScript. Hieronder volgt een voorbeeld

```
<div class="color-blue">
  <h2>This is a heading in a div element</h2>
  <p>This is some text in a div element.</p>
</div>
```

```
<div class="color-black">
  <h2>This is a heading in a div element</h2>
  <p>This is some text in a div element.</p>
</div>
```

## Nesting, lijsten en navigatiebalken

### Nesting

HTML elementen kunnen genest worden. Elk HTML document bestaat uit geneste elementen. Hieronder volgt een voorbeeld van een genest HTML document:

```
<!DOCTYPE html>
<html>
<body>

  <h1>My First Heading</h1>
  <p>My first paragraph.</p>
```

```
</body>
</html>
```

Geneste elementen zijn elementen die in elkaar zitten en de definitie komt vanuit het Engelse woord nesting ( objecten die in elkaar geplaatst kunnen worden ). Zo zie je dat tussen het container element <body> het element <h1> geplaatst is. Dit is een voorbeeld van nesting.

### Lijsten

Voor het onderdeel lijsten hebben we 3 soorten lijsten. een ul ( unordered list ) , een ol ( ordered list ) en een li ( een lijst element ).

Als eerst definiëren we wat voor soort lijst we willen, hier hebben we twee container element keuzes voor; een ul of een ol. Vervolgens kunnen we in dit element een li plaatsen/nesten. Hier volgen twee voorbeelden met de bijbehorende uitwerkingen:

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

- Coffee
- Tea
- Milk

---

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

1. Coffee
2. Tea
3. Milk

### Rondvraag:

Stel we zouden een sublijst willen maken met de volgende output, hoe nesten we dat?

1. Coffee
  1. Coffe wrong
  2. Coffe cappuchino
2. Tea
3. Milk

Antwoord:

<ol>

<li>Coffee</li>

<ol>

<li>Coffe wrong</li>

<li>Coffe cappuchino</li>

</ol>

<li>Tea</li>

<li>Milk</li>

</ol>

Navigatiebalken:

In HTML is er een <nav> tag dat gebruikt kan worden als een container tag voor een groep navigation links. Hieronder volgt een voorbeeld:

<nav>

<a href="/html/">HTML</a> |

<a href="/css/">CSS</a> |

<a href="/js/">JavaScript</a> |

<a href="/python/">Python</a>

</nav>

[HTML](#) | [CSS](#) | [JavaScript](#) | [Python](#)

De href in de link tag stuurt je door naar de baselink + href redirect. Wat hiermee bedoelt wordt is dat als je op [www.techgrounds.nl](http://www.techgrounds.nl) zit en je op de eerste link zou klikken je terecht zou komen op [www.techgrounds.nl/html](http://www.techgrounds.nl/html).

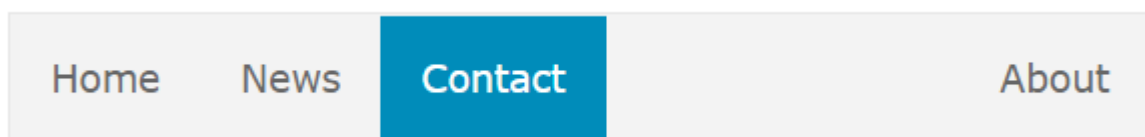
Het is niet gebruikelijk om navigatiebalken op deze wijze op te zetten. Voor screenreaders en slechtzienden kan dit element gebruikt worden om te besluiten of ze dit elementen willen renderen of niet. Dit element is niet essentieel voor veel webpagina's en screenreaders kunnen dit element dus weglaten.

Het is wel gebruikelijk om een navigatiebalk op te zetten door gebruik te maken van een lijst. Een navigatiebalk is simpelweg een lijst van links. Hieronder volgt een voorbeeld:

```
<ul>
  <li><a href="/home">Home</a></li>
  <li><a href="/news">News</a></li>
  <li><a href="/contact">Contact</a></li>
  <li><a href="/about">About</a></li>
</ul>
```

- [Home](#)
- [News](#)
- [Contact](#)
- [About](#)

Dit is hoe een standaard navigatiebalk opgezet wordt. Door middel van styling, CSS dat we in een later onderdeel zullen behandelen, kan je deze lijst stylen tot een mooie navigatiebalk:



## Opdrachten

### Image, audio & video

Het plaatsen van foto's op een website is een belangrijke toevoeging voor velen. Zo bestaan de meeste websites uit een grote collectie van foto's. Vaak kun je logo's, teamfoto's of andere soorten afbeeldingen terugvinden op een website. Het stylen van de afbeeldingen komt later in het onderdeel CSS aanbod.

1. Bij het inladen van een bestand is het noodzakelijk om de locatie van het bestand aan te geven. Dit kan op 2 manieren:
  - Absolute path
  - Relative path

Wat houden deze 2 paths in?

Een image element is voor het inladen van een afbeelding. Afbeeldingen kunnen van verschillende formaat zijn. Bijvoorbeeld, een afbeelding kan een JPG formaat zijn.

2. Zoek uit welke formaat afbeeldingen er nog meer kan zijn.

Een afbeelding tonen is afhankelijk van of de browser het ondersteunt.

3. Zoek uit welke browsers er zijn en maak een lijst van je antwoord.
4. Onderzoek hoe het plaatsen van een foto gedaan wordt en plaats deze boven en onderaan je website door zowel gebruik te maken van de absolute path en relative path.



#### Mijn eerste website

##### Mijn eerste sub-kop

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras sed pulvinar nisl, volutpat vulputate lectus. Morbi eget dapibus lectus. Fusce neque nunc, tempor et sapien eu, faucibus placerat nibh. Mauris nec sem mattis, congue libero eu, tincidunt nunc. Suspendisse potenti. Sed eget pellentesque sapien. Maecenas rutrum, augue eget dapibus feugiat, mauris arcu volutpat felis, vel tincidunt ipsum orci ut erat. Nulla auctor enim semper congue vestibulum.

##### Mijn tweede sub-kop

Nam henderit augue sagittis, elementum risus eu, pretium risus. Cras malesuada quam at sapien pellentesque elementum. Aliquam ipsum erat, faucibus et leo id, dapibus ultrices metus. Nunc porta magna non imperdiet tristique. Nullam congue ornare ante. In convallis felis non magna semper pharetra. Suspendisse dapibus elementum lacus, in tempor leo lobortis in. Proin semper risus tristique, tempus odio sit amet, dignissim magna. In non mi vehicula, auctor mi vel, lacinia lectus.

Suspendisse potenti. Nam sit amet magna leo. Pellentesque dignissim, metus id pulvinar malesuada, sapien magna vestibulum erat, non elementum lacus ligula ac est. In a feugiat tellus, in interdum mauris. Fusce pharetra eros quis dapibus fringilla. Integer a neque vitae sapien porttitor lobortis vitae ac arcu. Sed eu egestas augue, quis pellentesque dui. Praesent eu diam eu risus fringilla cursus. Donec a orci vitae nibh ullamcorper tristique. Nunc auctor mi ut elit rutrum placerat. Cras at venenatis nibh.

#### Footer

Deze website is gebouwd vanuit de [TechGrounds Academy](#)

Aan de img element kan je ook de *alt* attribute gebruiken.

5. Wat houdt de *alt* attribute in? En waarom is het handig om de *alt* attribuut te gebruiken?

In de sound folder staat een explosion.mp3 bestand.

6. Laadt dit bestand in als een audio op het scherm.

*Extra: Maak het leuker door de audio op repeat te zetten.*

*Tip: gebruik controls attribute om audio af te spelen.*

Een audio bestand heeft verschillende formaten. Niet alle formaten worden ondersteund door de browser. Hiervoor is een oplossing voor.

7. Speel een audio bestand af met <source> tag. Leg uit wat hier het voordeel van is.
8. Zoek een willekeurig video en sla dit op in je computer. Laadt dit bestand in als een video op het scherm.

*Tip: De video is veel groter dan het scherm. Zoek uit welke attribuut je voor de video element kan gebruiken om de grootte van de video aan te passen.*

Een video bestand heeft verschillende formaten. Niet alle formaten worden ondersteund door de browser. Hiervoor is een oplossing voor.

Speel een video bestand af met <source> tag. Leg uit wat hier het voordeel van is.

## Lijsten

Door gebruik te maken van lijsten in HTML kan je opsommingen eenvoudig weergeven. Onderzoek de verschillende soorten lijsten en op welke wijze je deze kunt toepassen.

9. Maak gebruik van de verschillende soorten lijsten om het volgende resultaat op je webpagina weer te geven.

1. item 1
2. item 2
3. item 3

1. item 1
2. item 2
  1. sub item 1
  2. sub item 2
3. item 3

- item 1
- item 2
- item 3

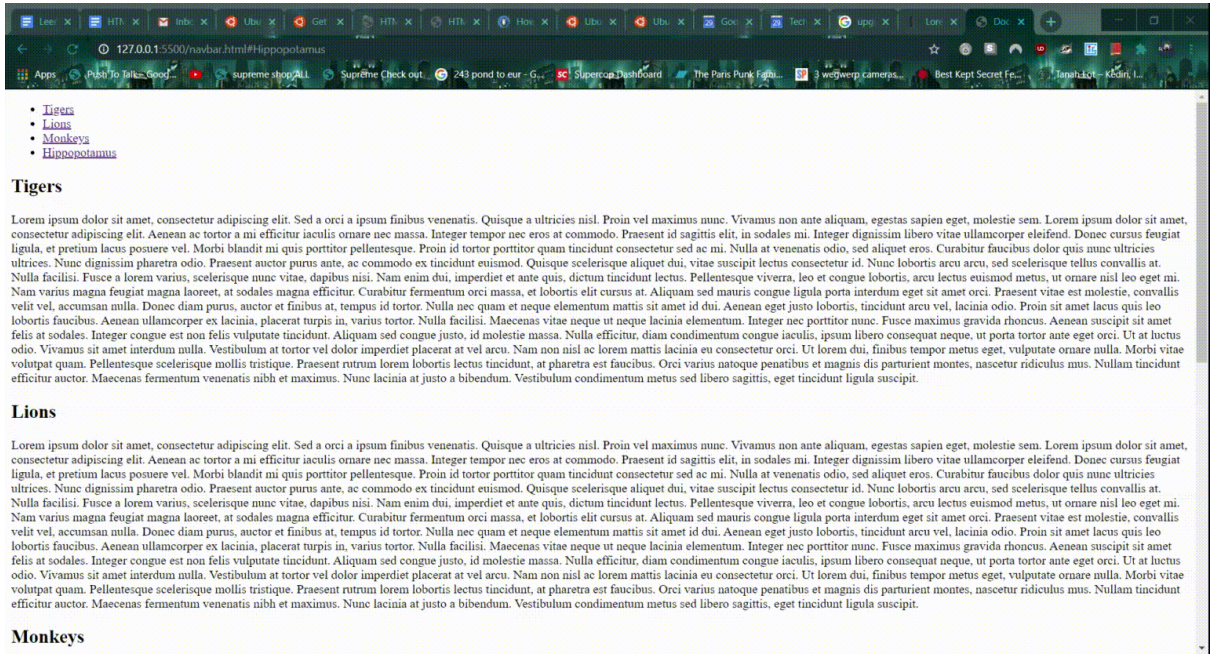
- item 1
- item 2
  - sub item 1
  - sub item 2
- item 3

- item 1
- item 2
  1. sub item 1
  2. sub item 2
- item 3

10. Bouw een navigatiebalk door gebruik te maken uit een lijst dat bestaat uit de volgende onderdelen:

- Tigers
- Lions
- Monkeys
- Hippopotaumus
  - [Tigers](#)
  - [Lions](#)
  - [Monkeys](#)
  - [Hippopotamus](#)

11. Creëer nu voor elk onderdeel een header en een bijbehorende stuk tekst en zorg ervoor dat door te klikken op de links in de navigatiebalk je gerefereerd wordt naar het bijbehorende onderdeel. Zie hieronder een voorbeeld:



## Onderdeel 3 - Forms & Tabellen

### HTML Forms

Een HTML form wordt gebruikt om gebruikers input te verzamelen. Deze data kan vervolgens opgestuurd worden naar een server om de data te verwerken of de website kan gebaseerd op de ingevulde data acties verbinden.

Er zijn meerdere soorten HTML form elementen. We zullen enkele belangrijke elementen nu los behandelen.

#### *Input*

`<input>`

Het input element heeft een attribuut genaamd type. Door type te definiëren zal het input element op een andere wijze weergegeven worden. Hieronder volgen enkele voorbeelden:

`<form>`

`<input type="text">`

`</form>`

Bij dit voorbeeld zie je een leeg tekstveld.

Het is gebruikelijk om voor een tekstveld een label toe te voegen zodat de gebruiker weet waar de input voor nodig is. Zie hieronder nog een voorbeeld.

`<form>`

`<input type="radio" id="male" name="gender" value="male">`

`<label for="male">Male</label><br>`

`<input type="radio" id="female" name="gender" value="female">`

`<label for="female">Female</label><br>`

`<input type="radio" id="other" name="gender" value="other">`

`<label for="other">Other</label>`

`</form>`

- ☐ Male
- ☐ Female
- ☐ Other

Hier zien we een voorbeeld met "radio buttons". Een radio button laat een gebruiker één optie kiezen uit een gelimiteerde lijst van opties. We zien hier ook naast elke radio button een label staan.

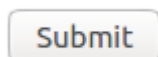


## Labels

Labels zijn erg handig voor gebruikers die gebruik maken van screenreaders. Door de labels zullen de labels hardop voorgelezen worden wanneer de gebruiker een input element highlight. Ook wordt door het klikken op een label de bijbehorende radio button/check-box geactiveerd. Het for attribuut van de label tag moet gelijk zijn aan het id attribuut van het input element zodat deze aan elkaar verbonden zijn en op een juiste wijze werken.

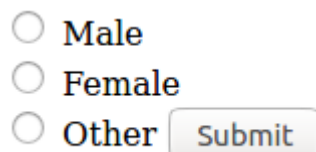
## Submit

```
<input type="submit">
```



Dit definieert een knop die de form data zal opsturen naar de gekozen locatie, dit noemen we de form-handler. Vaak is de form-handler een script die onze data zal verwerken. Het definiëren van een form-handler doen we als volgt:

```
<form action="/contact" method="POST">
<input type="radio" id="male" name="gender" value="male">
<label for="male">Male</label><br>
<input type="radio" id="female" name="gender" value="female">
<label for="female">Female</label><br>
<input type="radio" id="other" name="gender" value="other">
<label for="other">Other</label>
<input type="submit" value="Submit">
</form>
```



Let bij dit voorbeeld voornamelijk op de eerste regel. Hier zien we de actie en de method die bij de form tag horen. Dit is hoe wij de form-handler definiëren. We maken hier gebruik van een POST method, dit betekent dat we de data opsturen, die de data opstuurt naar een andere pagina op onze website. Op deze pagina zouden wij, om het niet al te ingewikkeld te maken en in details te gaan, een server kunnen verbinden die de data opvangt en voor ons opslaat.

## Verskil tussen GET en POST

Het verschil tussen de GET en POST methods is dat bij het gebruiken van de GET method de resultaten zichtbaar worden in de pagina's url. Dit is dus niet handig voor gevoelige data en verder heeft het ook een limiet in grootte. Voor niet gevoelige data is het gebruiken van een GET method wel handig. Op deze wijze wordt de input in een google search-bar meegestuurd. Als je een input hebt die erg groot kan zijn of gevoelige data kan bevatten maken we gebruik van de POST method.

In onze voorbeelden kan teruggezien worden dan elk element dat we willen opsturen een name attribuut heeft. Als een name attribuut niet wordt meegegeven, wordt de data van het input veld niet opgestuurd. Onthoud dit dus altijd.

## HTML Form elementen

Binnen de form container tags kunnen de volgende form elementen geplaatst worden:

- `<label>`
- `<input>`
- `<select>`
- `<textarea>`
- `<button>`
- `<fieldset>`
- `<legend>`
- `<datalist>`
- `<output>`
- `<option>`
- `<optgroup>`

Het is van belang dat er zelfonderzoek gedaan wordt naar hoe deze elementen gebruikt worden, wat de verschillen tussen deze elementen zijn en welke attributen meegegeven moeten worden om het gewilde resultaat te bereiken.

## HTML Form Input types

Er zijn veel verschillende soorten input types. Enkele hebben wij al behandeld, hieronder volgt een volledige lijst van de input types:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`

- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

Dit onderdeel zal duidelijk worden tijdens het maken van de opgaven. Het is niet van belang alle input types en de attributen te onthouden. Wel van belang is het juist opzoeken en googlen naar het gewilde input type. Hier zullen we meer mee oefenen in de opgaven.

## HTML Tabellen

HTML tabellen zorgen ervoor dat de codeur data kan groeperen in rijen en kolommen. Hieronder volgt een voorbeeld:

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

De <table> container tag definieert een HTML tabel. Hierin plaatsen wij de rijen, <tr>, die bijbehorende headers hebben, <th>, en als laatst wordt elke tabelcel gedefinieerd door de <td> tags. Hieronder volgt een voorbeeld:

```
<table>
<tr>
  <th>Firstname</th>
  <th>Lastname</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
</table>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

Hier zien we dat we 3 kolommen hebben. Deze kolommen zijn Firstname, lastname en age. Vervolgens heeft elke rij, de <td> tag, 3 ingevulde waardes. Deze waardes zijn

respectievelijk de voornaam, achternaam en leeftijd. Voor elke nieuwe toevoeging maken we een nieuwe rij aan (<tr>).

Om een titel toe te voegen aan onze tabel voegen wij een <caption> tag toe. Dit ziet er als volgt uit:

```
<table>
<caption>Info avond data</caption>
<tr>
  <th>Firstname</th>
  <th>Lastname</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
</table>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

### *Groeperen van onze tabellen*

Tabellen kan je groeperen net als dat we andere onderdelen van onze webpagina groeperen. Zo zijn er 3 groepering elementen. thead, tbody en tfooter. thead gebruiken wij om onze headers te groeperen, tbody om alle basis content te groeperen en tfooter voor de content die we als footer content zien. Als voorbeeld kunnen we alle headers in thead stoppen, alle basic data in tbody en onderdelen zoals opsommingen/totaal in onze tfooter. Hieronder volgt een voorbeeld:

```
<table>
<thead>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
</thead>
<tbody>
<tr>
```

```
<td>January</td>
<td>$100</td>
</tr>
<tr>
<td>February</td>
<td>$80</td>
</tr>
</tbody>
<tfoot>
<tr>
<td>Sum</td>
<td>$180</td>
</tr>
</tfoot>
</table>
```

Month	Savings
January	\$100
February	\$80
Sum	\$180

Zoals gezien kan worden heeft dit geen invloed op de weergave van onze pagina. Wel is dit onze data beter in te lezen voor screenreaders en browsers voor slechtzienden.

## Opdrachten

### HTML Forms

HTML formulieren zijn de belangrijkste schakel tussen een gebruiker en een website of een applicatie. Zij laten gebruikers toe gegevens naar websites te sturen. Meestal zal dat naar de server zijn maar een webpagina kan de gegevens ook zelf interpreteren.

Een HTML form bestaat uit een aantal elementen. Bijvoorbeeld textfields (bestaande uit één of meerdere regels), choices of buttons.

1. Maak een HTML form waarbij de gebruiker zijn naam en achternaam in kan vullen. Zie hieronder voor een voorbeeld:



Achternaam:

Voornaam:

2. Naast input text fields zijn er ook nog andere input fields. Zoals:
- radiobutton

- checkbox
- password
- reset
- submit
- date
- search
- time
- hidden
- email

Maak voor elk input field een voorbeeld.

3. Maak een input email field met een submit knop.

*tip: wat gebeurt er wanneer je op de submit knop clickt met een random text?*

4. Een formulier kun je ook groeperen. Zie hieronder voor een voorbeeld:



Persoons gegevens

Achternaam:

Voornaam:

Geboortedatum

M: ☐ V: ☐

Postcode:

Adres:

Bouw dit voorbeeld na.

5. Naast input element heb je ook een select element. Hiermee kan je een dropdown list maken. Maak een drop down list.
6. We hebben nu geleerd hoe je dropdown list, radio buttons en checkboxes maak. Dit zijn 3 verschillende manieren om de gebruiker een keuze te laten maken. Benoem voor dropdown list, radio buttons en checkboxes een situatie waar je in kan gebruiken en leg dat uit?
7. Maak een reserveringsformulier om een sport te reserveren. In dit formulier willen we graag het volgende weten:
  - Naam van dat persoon
  - Off hij/zij lidmaatschap heeft
  - E-mail van dat persoon

- Type sport om te reserveren. Keuze uit: tafeltennis, badminton of squash
  - Datum en tijdstip om te reserveren
  - Een checkbox waarbij de gebruiker akkoord gaat over de reserveringsvoorwaarden
  - Een submit knop
8. Maak een reserverings formulier om een boek te reserveren. In dit formulier willen we graag het volgende weten:
- Naam van dat persoon
  - Telefoonnummer
  - E-mail van dat persoon
  - Naam van dat boek
  - Aantal boeken om te reserveren. Het aantal moet minimaal 1 en maximaal 5 zijn.
  - Een submit knop
9. Bouw nu zelf een formulier met wat je hiervoor hebt geleerd.

## HTML tables

10. Om een tabel te maken heb je volgende elementen nodig:

- `<table>`
- `<thead>`
- `<tbody>`
- `<tfoot>`
- `<th>`
- `<tr>`
- `<td>`

Geef voor elke element een definitie.

11. Maak een tabel met `<table>` element. Zie hieronder:

Voornaam	Achternaam	Leeftijd	Telefoonnr
John	Doe	41	+31 12345678
Steven	Smith	32	+31 12345678
Alexander	Beave	16	+31 12345678

12. Maak een agenda tabel met `<table>` element. Zie hieronder:

MA	DI	WO	DO	VR
Afspraak met tandarts				Sporten
	Huiswerk maken	Werken	Werken	
Afspraak met tandarts			Werken	Bij vrienden



13. Voor het maken van een tabel kan je ook gebruik maken van colspan en rowspan attributen. Zoek uit wat colspan en rowspan attribuut doet en meet het volgende:

	Sport	Aantal recreanten	Aantal competitie spelers	Totale deelnemers
Racket sport	Badminton	10	30	40
	Tafeltennis	59	30	89
	Tennis	159	130	189
Tennis	159	130	189	
Ball sport	Voetbal	88	40	128
	Basketbal	60	40	100
	Volleybal	35	20	55

14. Maak nu zelf een tabel met de elementen die je hiervoor hebt geleerd.

*Maak geen gebruik van css.*