

Bootstrap workshop

1 Wat is Bootstrap?

Bootstrap is een CSS Framework om op een eenvoudige en snelle manier een responsive website te maken. Er zijn twee manieren om bootstrap te gebruiken. Werken met de compiled CSS en JS version of werken met de source Sass en JavaScript files.

Compiled version

Als je werkt met de compiled CSS version van Bootstrap dan kan je Bootstrap zien als een .css-file (eventueel aangevuld met een JavaScript-file) die door iemand anders gemaakt is en die jij moet leren gebruiken.

Om je webpagina's op te maken met behulp van de compiled version van Bootstrap, ga je dus zelf weinig of geen CSS-regels schrijven, maar ga je gewoon classes (die al aanwezig zijn in de bootstrap CSS-file) toevoegen aan je HTML-pagina's.

Source files

Het is ook mogelijk om met de .SCSS source files te werken en zelf het framework aan te passen.

In deze workshop zullen we werken met de Compiled version. Deze workshop is gebaseerd op de [documentatie van bootstrap](#).

Een alfabetische lijst van Bootstrap classes, waarbij de lijst ook te sorteren is op Category vind je op [w3schools Bootstrap 4 Class Reference](#).

2 Bootstrap downloaden

1. Open de map **bootstrap-workshop** in Visual Studio Code.
2. De beschrijving van wat elk van de twee downloads (**Precompiled Bootstrap** en **Bootstrap source code**) bevat vind je op <https://getbootstrap.com/docs/4.5/getting-started/contents/>
Download de compiled version van Bootstrap (<https://getbootstrap.com/docs/4.5/getting-started/download/>) en unzip de file. Je bekomt onderstaande structuur:

```
bootstrap/
├── css/
│   ├── bootstrap-grid.css
│   ├── bootstrap-grid.css.map
│   ├── bootstrap-grid.min.css
│   ├── bootstrap-grid.min.css.map
│   ├── bootstrap-reboot.css
│   ├── bootstrap-reboot.css.map
│   ├── bootstrap-reboot.min.css
│   ├── bootstrap-reboot.min.css.map
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   └── bootstrap.min.css.map
└── js/
    └── bootstrap.bundle.js
```

```

├── bootstrap.bundle.js.map
├── bootstrap.bundle.min.js
├── bootstrap.bundle.min.js.map
├── bootstrap.js
├── bootstrap.js.map
├── bootstrap.min.js
└── bootstrap.min.js.map

```

Er zijn drie Precompiled Bootstrap versies nl. **bootstrap.css**, **bootstrap-grid.css** en **bootstrap-reboot.css**.

Voor elke versie is er ook een [minified](#) CSS (en JS) (`bootstrap.min.*`). Er zijn ook [source maps](#)-files (`bootstrap.*.map`) beschikbaar die ervoor zorgen dat je de `.scss` code kan zien in sommige browser developer tools.

3. Wij gaan gebruikmaken van de file **bootstrap.css**. Kopieer **bootstrap.css** naar de **css**-map. Normaal zou je de minified versie **bootstrap.min.css** gebruiken, omdat deze kleiner is, maar om te leren werken met Bootstrap is het handig om af en toe eens de **bootstrap.css**-file te kunnen bestuderen.
4. Open in Visual Studio Code het bestand **containers.html** en leg een link naar het bestand **css/bootstrap.css**. Om bootstrap te kunnen gebruiken volstaat dit momenteel. Een link naar de JavaScript file zullen we later leggen als we die nodig hebben.

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Containers</title>
  <link rel="stylesheet" href="css/bootstrap.css">
</head>

```

3 Responsive breakpoints

Bootstrap is ontworpen volgens het 'Mobile first' concept en maakt gebruik van Media queries.

Bijgevolg is de viewport meta tag noodzakelijk voor de goede werking van de CSS-code:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

De bootstrap css-file gebruikt onderstaande `@media`-query's (breakpoints) en in de class-names van bootstrap worden deze breakpoints steeds afgekort als:

sm: small devices; md: medium device, lg: large devices en xl: extra large devices

```

// Extra small devices (portrait phones, less than 576px)
// No media query for `xs` since this is the default in Bootstrap

// Small devices `sm` (landscape phones, 576px and up)
@media (min-width: 576px) { ... }

// Medium devices `md` (tablets, 768px and up)
@media (min-width: 768px) { ... }

// Large devices `lg` (desktops, 992px and up)
@media (min-width: 992px) { ... }

```

```
// Extra large devices `xl` (large desktops, 1200px and up)
@media (min-width: 1200px) { ... }
```

Extra info: als je werkt met de .SCSS Sources files dan kan je deze breakpoints aanpassen door de variabelen `$grid-breakpoints` en `$container-max-widths`, die sass maps bevatten, te overschrijven en Bootstrap opnieuw te compileren. Meer info over hoe je dit best doet vind je op de bootstrap website: [Theming Bootstrap](#).

<pre>// Grid breakpoints // // Define the minimum dimensions at which // your layout will change, // adapting to different screen sizes, for // use in media queries. \$grid-breakpoints: (xs: 0, sm: 576px, md: 768px, lg: 992px, xl: 1200px) !default;</pre>	<pre>// Grid containers // // Define the maximum width of `.container` // for different screen sizes. \$container-max-widths: (sm: 540px, md: 720px, lg: 960px, xl: 1140px) !default;</pre>
---	--

4 Containers

1. Bekijk **containers.html** in je browser en merk op dat er standaard geen padding is.

Om de webpagina van enige padding te voorzien en om eventueel de inhoud te centreren moet je een bootstrap-container gebruiken.

Bootstrap bevat drie soorten containers:

- de class `.container` stelt een `max-width` in bij elk responsive breakpoint
 - `.container-fluid`, is `width: 100%` bij alle breakpoints
 - `.container-{breakpoint}`, is `width: 100%` tot het opgegeven breakpoint en stelt daarna een `max-width` in bij elk volgend break-point.
- Zo heeft `.container-lg` een `width:100%` zolang de viewport kleiner is dan `992px`. Vanaf `992px` krijgt de container een vaste breedte van `960px` en vanaf `1200px` wordt deze vaste breedte `1140px`.

2. Voeg een div-element met class `container` toe aan **containers.html**:

```
<body>
<div class="container">
  <h1>Containers</h1>
  <p>
    Containers are the most basic layout element in Bootstrap and are
    <strong>required when using the default grid system</strong>.
  </p>
```

Containers are used to contain, pad, and (sometimes) center the content within them. While containers can be nested, most layouts do not require a nested container.

```
</p>
<hr>
</div>
</body>
```

De webpagina ziet er nu met Bootstrap grafisch een stuk beter uit dan met de standaard browseropmaak (zie ook onderstaande schermafbeeldingen). Er is een ander lettertype ingesteld, er is meer witruimte: o.a. de regelhoogte is aangepast. Ook de horizontale lijn is subtieler gekleurd en de lettertypekleur is geen zwart meer maar een grijsstint.

Containers

Containers are the most basic layout element in Bootstrap and are **required when using the default grid system**. Containers are used to contain, pad, and (sometimes) center the content within them. While containers can be nested, most layouts do not require a nested container.

Figuur 1: containers.html met standaard browseropmaak

Containers

Containers are the most basic layout element in Bootstrap and are **required when using the default grid system**. Containers are used to contain, pad, and (sometimes) center the content within them. While containers can be nested, most layouts do not require a nested container.

Figuur 2: containers.html met bootstrap

3. Voorzie het container-element van een achtergrondkleur om beter te zien wat er gebeurt als je het browservenster versmalt en verbreedt. We gebruiken hier even een style-element omdat dit vlugger werkt dan een aparte .css-file gebruiken.

```
...
<style>
  .container {
    background-color: bisque;
  }
</style>
</head>

<body>
  <div class="container">
    <h1>Containers</h1>
    <p>
      Containers are the most basic layout element in Bootstrap and are
      <strong>required when using the default grid system</strong>.
      Containers are used to contain, pad, and (sometimes) center the content
      within them. While containers can be
      nested, most layouts do not require a nested container.
    </p>
    <hr>
  </div>
```

</body>

Versmal en verbreed vervolgens je browservenster en controleer dat bij elk breakpoint de `max-width` verandert. Bij elk breakpoint verspringt de containerbreedte .

Test ook eens `.container-fluid` en bijvoorbeeld `.container-lg` uit.

BRON: <https://getbootstrap.com/docs/4.5/layout/overview/#containers>

4. Zoek in **bootstrap.css** naar de verschillende css-declaraties voor `.container`, `.container-fluid`, `.container-sm`, ...

Je zal onderstaande CSS-regels vinden in **bootstrap.css**. Met je kennis van CSS zou je deze regels moeten kunnen begrijpen en begrijp je nu waarom de containerbreedte telkens verspringt als je jouw browservenster smaller en breder maakt.

```
.container,
.container-fluid,
.container-sm,
.container-md,
.container-lg,
.container-xl {
  width: 100%;
  padding-right: 15px;
  padding-left: 15px;
  margin-right: auto;
  margin-left: auto;
}

@media (min-width: 576px) {
  .container, .container-sm {
    max-width: 540px;
  }
}

@media (min-width: 768px) {
  .container, .container-sm, .container-md {
    max-width: 720px;
  }
}

@media (min-width: 992px) {
  .container, .container-sm, .container-md, .container-lg {
    max-width: 960px;
  }
}

@media (min-width: 1200px) {
  .container, .container-sm, .container-md, .container-lg, .container-xl {
    max-width: 1140px;
  }
}
```

5 Bootstrap Grid System

Het bootstrap grid system gebruik containers, rows en columns. Elk van deze drie class-types: `.container*`, `.row` en `.col*` is noodzakelijk voor de goede werking van het bootstrap grid system. Bovendien mag een element met `class="row"` enkel elementen met `class="col*"` bevatten. Het voorbeeld op de bootstrap website ziet er als volgt uit:



The image shows a visual representation of the Bootstrap grid system. At the top, there are three light purple rectangular boxes, each containing the text "One of three columns". Below these boxes is a code editor with the following HTML code:

```
<div class="container">
  <div class="row">
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
  </div>
</div>
```

To the right of the code editor is a small blue button labeled "Copy".

Dit voorbeeld start met één kolom op een extra small device en creëert drie kolommen van gelijke breedte op small, medium, large en extra large devices en deze kolommen worden gecentreerd op de pagina met `.container`.

We zullen nu zelf een voorbeeld maken door het bestand **grid-system.html** aan te passen.

Op een extra small en small device willen we geen kolommen en moeten de div-elementen onder elkaar worden weergegeven (we starten **Stacked**). Op een medium, large en extra large device moeten de div-elementen naast elkaar worden weergegeven (**horizontal**). Hierbij moet de eerste div dubbel zo breed zijn als de tweede div.

Bootstrap werkt met een grid-system van **12-kolommen**. In het onderstaande voorbeeld gebruikt, vanaf het medium breakpoint, de eerste div **acht** kolommen en de tweede div gebruikt **vier** kolommen.

Bootstrap Grid system

Heading

Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui.

Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.

Heading

Donec sed odio dui. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.

1. Open **grid-system.html** en voeg een link toe naar **css/bootstrap.css**.

Pas vervolgens de HTML als volgt aan:

```
<div class="container">
  <h1>Bootstrap Grid system</h1>
  <!-- Example row of columns -->
  <div class="row">
    <div class="col-md-8">
      <h2>Heading</h2>
      <p>Donec id elit non mi porta gravida at eget metus. Fusce dapibus,
        tellus ac cursus commodo, tortor mauris
        condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta
        sem malesuada magna mollis euismod.
        Donec sed odio dui. </p>
      <p>Donec id elit non mi porta gravida at eget metus. Fusce dapibus,
        tellus ac cursus commodo, tortor mauris
        condimentum nibh, ut fermentum massa justo sit amet risus. </p>
    </div>
    <div class="col-md-4">
      <h2>Heading</h2>
      <p>Donec sed odio dui. Cras justo odio, dapibus ac facilisis in, egestas
        eget quam. Vestibulum id ligula porta
        felis euismod semper. Fusce dapibus, tellus ac cursus commodo, tortor
        mauris condimentum nibh, ut fermentum
        massa justo sit amet risus.</p>
    </div>
  </div>
</div>
```

2. Controleer door je browservenster te versmallen en te verbreden of alles werkt zoals verwacht.
3. Het Bootstrap Grid system maakt geen gebruik van Grid Layout maar gebruikt Flex Layout. Elk element met class **row** is een flex-container en de elementen met class **col*** zijn flex-items.

Controleer met behulp van de Developer Tools dat het div-element met class `row` een flex-container is.

4. We kunnen ook meerdere classes opgeven. Om vanaf Large devices twee kolommen van gelijke breedte te creëren voeg je aan beide div-elementen nog de class `col-lg-6` toe:

```
<div class="col-md-8 col-lg-6">  
  
<div class="col-md-4 col-lg-6">
```

Controleer opnieuw door je browservenster te versmallen en te verbreden.

Merk op dat twee keer `col-lg` gebruiken in plaats van `col-lg-6` ook volstaat om twee gelijke kolommen te maken.

In het bovenstaande voorbeeld zijn er div-elementen gebruikt. Misschien nog even herhalen dat je deze div-elementen ook altijd kan vervangen door logische blokken met een semantische betekenis. De twee div-elementen zouden bijvoorbeeld evengoed een main en een aside-element kunnen geweest zijn. Of als de twee kolommen onafhankelijke artikels bevatten, twee article-elementen. Dit alles heeft logischerwijs geen enkele invloed op het uiteindelijk visueel resultaat.

Voor meer info zie : <https://getbootstrap.com/docs/4.5/layout/grid/>

of w3schools.com https://www.w3schools.com/bootstrap4/bootstrap_grid_system.asp

6 Content

6.1 Reboot

Bootstrap maakt gebruik van een reboot-file die gebaseerd is op de **normalize.css** file van Nicolas Gallagher die we gebruikt hebben in de vanderbeken-oefening. Je kan dit controleren door **bootstrap-reboot.css** eens te openen in Visual Studio Code.

Meer info over reboot: <https://getbootstrap.com/docs/4.5/content/reboot/>

6.2 Images

Van de beschikbare classes voor images bespreken we enkel `.img-fluid`.

1. Open het bestand **images.html** en voeg een link toe naar **css/bootstrap.css**.

Je ziet dat de afbeeldingen niet responsive zijn. Als we zelf onze CSS zouden schrijven dan zouden we de afbeelding responsive maken met (minimaal) de volgende .css-declaratie:

```
max-width: 100%;
```

eventueel aangevuld met `height: auto;` als er een expliciete hoogte ingesteld is in HTML.

2. In bootstrap gaan we echter ook hier terug geen CSS schrijven maar moet je de klasse `img-fluid` toekennen aan de afbeeldingen. Voer dit uit en controleer in je browser dat de afbeeldingen responsive geworden zijn.

Images

Heading

Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui.



Heading

Donec sed odio dui. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.



3. Zoek ook nog eens de klasse `.img-fluid` op in **bootstrap.css**.

Meer info over Images-classes: [Content > Images](#)

7 Components

7.1 Buttons

Buttons zitten in de documentatie van bootstrap onder Components ([Components > Buttons](#)).

Examples

Bootstrap includes several predefined button styles, each serving its own semantic purpose, with a few extras thrown in for more control.



Voor het opmaken van components werken de makers van bootstrap meestal met meerdere classes. Een base-class en modifier-classes. Voor buttons is de base-class `.btn` en zijn er verschillende modifier-classes: `.btn-primary`, `.btn-lg`, ...

De `.btn`-classes zijn in eerste instantie ontworpen om gebruikt te worden met het `<button>`-element.

1. Open **buttons.html** en voeg een link toe naar **css/bootstrap.css**.

2. Maak vervolgens van het eerste `<button>`-element een Large Primary button en van het tweede `<button>`-element een Dark button.

```
<p>
  <button type="button" class="btn btn-primary btn-lg">Large button</button>
  <button type="button" class="btn btn-dark">Dark button</button>
</p>
```

7.2 Collapse

Om in **components-utilities.html** de Lorem ipsum-tekst te verbergen en pas weer terug weer te geven als er geklikt wordt op de knop, kunnen we de Bootstrap Collapse-component gebruiken ([Components > Collapse](#)).

De tot nu toe besproken bootstrap-items maakten enkel gebruik van CSS. De Collapse-component maakt echter ook gebruik van JavaScript, bijgevolg moeten we om deze component te kunnen gebruiken ook een link naar de Bootstrap JavaScript file toevoegen.

In plaats van een link toe te voegen naar **bootstrap.bundle.min.js** zullen we in dit voorbeeld echter een [CDN \(Content Delivery Network\)](#) gebruiken. De link en script-elementen die je hiervoor nodig hebt kan je kopiëren vanuit de [starter-template](#) op de bootstrap-website of vanuit het voorbeeld hieronder. Het gebruikte CDN is [JSDELIVR](#), een gratis CDN voor Open Source projecten.

```
<!DOCTYPE html>
<html lang="nl">

<head>
<meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Hello, World!</title>

  <!-- Bootstrap CSS -->
  <link rel="stylesheet"
    href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"
    integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2"
    crossorigin="anonymous">
</head>

<body>
  . . .

  <!-- Option 1: jQuery and Bootstrap Bundle (includes Popper) -->
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
    integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
    crossorigin="anonymous"></script>
  <script
    src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-ho+j7jyWK8fNQe+A12Hb8AhRq26LrZ/JpcUGGOn+Y7RsweNrtn/tE3MoK7ZeZDyx"
    crossorigin="anonymous"></script>
</body>
</html>
```

Als je de voorbeelden van de Collapse component op de Bootstrap website bestudeert dan zal je zien dat deze voorbeelden **data-*** en **aria-*** attributen bevatten.

Als je in HTML zelf een attribuut wilt definiëren dan moet je de naam van het attribuut laten beginnen met **data-**. Zo hebben de makers van bootstrap twee eigen attributen gedefinieerd.

data-toggle="collapse" definieert de gewenste functionaliteit en **data-target="#collapseExample"** bevat het id (eigenlijk de CSS selector) van het element dat moet dichtgeklapt (collapsed) worden.

Beide attributen zijn nodig om de Bootstrap JavaScript-code, die bij de Collapse component hoort, correct te laten werken.

Daarnaast bevatten de voorbeelden ook **aria-*** attributen. Zoals de **data-*** attributen bedoeld zijn om de Bootstrap-scripts informatie te verschaffen, zo zijn de **aria-*** attributen bedoeld om screen-readers meer informatie te verschaffen.

Ter info: de **data-*** en **aria-*** HTML-attributen worden in het boek kort aangehaald op pagina 39 en 42.

1. Open **components-utilities.html** en voeg (tenzij je dit al hebt uitgevoerd) de nodige verwijzingen toe naar bootstrap (maak gebruik van het gratis CDN JSDELIVR).
2. Voeg nu de Collapse component toe aan **components-utilities.html**. Het paragraaf-element moet je in een div-element 'wrappen' en je moet de **data-*** en **aria-***-attributen toevoegen aan de button.

```
<h1>'Button' met 'Collapse component'</h1>
<p><button type="button" class="btn btn-primary btn-lg"
  data-toggle="collapse" data-target="#collapseContainer"
  aria-expanded="false" aria-controls="collapseContainer">De 'lorem ipsum' tekst
  weergeven/verbergen</button></p>
```

```
<div class="collapse" id="collapseContainer">
  <p>Lorem ipsum dolor sit amet consectetur
    adipiscing elit. Rerum facilis id earum eius possimus!
    Quam, accusamus placeat commodi iure tempore sequi deserunt dicta
    distinctio esse, vero nobis architecto ad fuga!</p>
</div>
```

3. Test in je browser of de knop werkt.

8 Utilities

Bootstrap bevat een groot aantal Utility classes voor het toevoegen van kleuren, borders, witruimte, enz. aan HTML-elementen.

8.1 Borders

4. Voeg in **components-utilities.html** een afgeronde border toe aan de paragraaf. Meer info over de Borders Utility classes vind je op ([Utilities > Borders](#)).

```
<p class="border rounded">Lorem ipsum dolor sit amet consectetur
```

8.2 Spacing

5. Voeg in **components-utilities.html** wat padding toe aan de paragraaf, zie ([Utilities > Spacing](#)) en controleer het resultaat in jouw browser.

```
<div class="collapse" id="collapseContainer">
  <p class="border rounded p-2">Lorem ipsum dolor sit amet consectetur
    adipiscing elit. Rerum facilis id earum eius possimus!
    Quam, accusamus placeat commodi iure tempore sequi deserunt dicta
    distinctio esse, vero nobis architecto ad fuga!</p>
</div>
```

9 Navbar Component

We eindigen deze workshop met een voorbeeld van de navbar-component ([Component > Navbar](#))

1. Start alvast met in de **css**-map een bestand **main.css** aan te maken. Dit omdat we in dit voorbeeld zelf een CSS-regel gaan toevoegen om de achtergrondkleur van het nav-element in te stellen. Als we met bootstrap werken, gaan we zoveel mogelijk de classes van bootstrap gebruiken, maar soms is er geen class die doet wat we wensen en dan moeten we zelf CSS-regels schrijven (of de SASS Source code aanpassen).
2. Open het bestand **navbar.html** en voeg de nodige verwijzingen toe naar bootstrap (maak gebruik van het gratis CDN JSDELIVR). De navbar-component (met hamburger-icon) maakt gebruik van de Collapse-component en maakt dus ook gebruik van JavaScript. Voeg ook een link tot naar jouw css-bestand **css/main.css**.

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Navbar example</title>

  <!-- Bootstrap CSS -->
  <link rel="stylesheet"
    href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"
    integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2"
    crossorigin="anonymous">
  <link href="css/main.css" rel="stylesheet">
</head>

<body>

  . . .

  <footer class="container-md">
    <p>&copy; Company 2017-2020</p>
  </footer>
```

```

<!-- Option 1: jQuery and Bootstrap Bundle (includes Popper) -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
  integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
  crossorigin="anonymous"></script>
<script
  src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-ho+j7jyWK8fNQe+A12Hb8AhRq26LrZ/JpcUGGOn+Y7RsweNrtn/tE3MoK7ZeZDyx"
  crossorigin="anonymous"></script>
</body>
</html>

```

De documentatie van Bootstrap is niet altijd even uitgebreid. Er wordt niet altijd duidelijk beschreven welke classes en attributen minimaal noodzakelijk zijn om iets te laten werken en wat de functie van elke class en attribuut is. Meestal zal je dus werken naar analogie met een voorbeeld op de website.

Hieronder vind je een aangepaste versie van het eerste navbar-voorbeeld op de bootstrap-website ([Components > Navbar](#)).

De navbar-component in het voorbeeld bevat de volgende sub-components:

- `.navbar-brand`: voor het logo
- `.navbar-toggler`: de knop met het hamburger-icon (het span-element `` zorgt voor het hamburger-icon).
- `.navbar-nav`: de navigatiebalk
- `.collapse.navbar-collapse` bevat de te verbergen elementen

Het nav-element `<nav class="navbar navbar-expand-md navbar-light">` bevat een base-class `navbar` en twee modifier-classes:

- `navbar-expand-md` zorgt voor 'responsive collapsing' bij het medium breakpoint.
- `navbar-light` zorgt voor een donker lettertype. Voor gebruik met een lichte achtergrondkleur.

Er is aan het voorbeeld een container-element `<div class="container-md">` toegevoegd zodat de navbar, net als de overige elementen in de webpagina, vanaf het breakpoint Medium, een vaste breedte krijgt en gecentreerd wordt.

De Navbar Component gebruikt de Collapse component en bevat dus dezelfde **data-*** en **aria-*** attributen als deze component.

Bij **navbar-nav** wordt de linkermarge met de **ml-auto** utility class ingesteld op **auto**, zodat de flex-layout dit element naar rechts duwt.

```

<nav class="navbar navbar-expand-md navbar-light">
  <div class="container-md">
    <a class="navbar-brand" href="#"></a>
    <button class="navbar-toggler" type="button" data-toggle="collapse"
      data-target="#navbarsExampleDefault"
      aria-controls="navbarsExampleDefault" aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>

```

```

</button>

<div class="collapse navbar-collapse" id="navbarsExampleDefault">
  <ul class="navbar-nav ml-auto">
    <li class="nav-item active">
      <a class="nav-link" href="#">Home <span
        class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
    <li class=" nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
  </ul>
</div>
</nav>

```

3. Voeg bovenstaande HTML-code (het volledige nav-element) toe aan **navbar.html**.
4. In bovenstaande code hebben we geen gebruikgemaakt van de bootstrap color-utilities ([Utilities > Colors](#)) om de [achtergrondkleur](#) van de navbar-component in te stellen, omdat we onze eigen kleur willen gebruiken. Stel de kleur van het nav-element in door onderstaande stijlregel toe te voegen aan **css/main.css**.

```

nav {
  background-color: #e3f2fd;
}

```

5. Verbreed en versmal je browser-venster en controleer dat voor Small devices er een hamburger-menu is en dat vanaf het Medium breakpoint er een horizontaal menu is.