

ADVANCED HTML EN CSS

1. BOOTSTRAP, SASS, JAVASCRIPT EN PREPROS INSTALLATIE

Bootstrap is een mobile first ontwikkelingsframework om vlug responsive websites te kunnen ontwikkelen. De leercurve in het begin is stijl aangezien dat je alle componenten van het framework dient te kennen.

Bootstrap is modulair en bestaat hoofdzakelijk uit een reeks van SCSS die de verschillende onderdelen van de toolkit verwezenlijken.

Het tabelsysteem en responsive design heeft standaard een lay-out met een tabel van 1170 pixels breed. Als alternatief kan de ontwikkelaar kiezen voor een lay-out met variabele breedte. Voor beide gevallen biedt de toolkit vier variaties aan voor verschillende resoluties en apparaattypes.

SCSS bestanden zijn eigenlijk CSS bestanden waarin je met variabelen kan gaan werken. Deze bestanden worden omgezet door een preprocessor naar bootstrap.min.css.

De preprocessor die we in de cursus zullen gebruiken is prepros die te verkrijgen is voor zowel MAC als Windows.

Bootstrap bevat ook meerdere JavaScript-componenten in de vorm van jquery-plugin-ins. Deze zijn voorzien in bijkomende elementen voor de gebruikersinterface. Voorbeelden hiervan zijn bijvoorbeeld dialoogvensters, tooltips, carousels,.... Ze breiden ook de functionaliteit van bestaande elementen uit, zoals bijvoorbeeld de auto-complete voor invoervelden.

Dit is slechts een korte inleiding en het topje van de ijsberg binnen het Bootstrap Framework. Deze cursus zal jullie stap per stap begeleiden in het opzetten van een project. Deze opzet is dezelfde voor kleine, middelgrote en zelfs grote projecten.

1.1. SOORTEN INSTALLATIES

1.1.1. INSTALLATIE METHODE 1: CDN

Je kan Bootstrap gebruiken door gebruik te maken van CDN (=content delivery network). Dit kan je vergelijken met bijvoorbeeld de google webfonts die online staan. Zoals je reeds weet kan je dan via de juiste link in de header tag van een html-bestand de google fonts aanspreken. Om gebruik te maken van het bootstrap framework dienen we eigenlijk juist hetzelfde te doen, maar spreken we een CDN aan. Een CDN herbergt eigenlijk de volledige bootstrap code die we kunnen aanspreken.

Hoe dienen we dit nu te implementeren.

1. Ga naar de site van Bootstrap, nl. www.getbootstrap.com
2. Vervolgens zie je de rubriek Bootstrap CDN staan. Bootstrap heeft zijn eigen css link die je dient te kopiëren en in je header te verwerken van je webpagina.
3. Daarnaast komen we in aanraking met javascript. Er zijn 3 bibliotheken (libraries) waarvan Bootstrap gebruik maakt, nl. bootstrap, jquery en popper. Deze laatste is nieuw sinds Bootstrap 4.



Bootstrap CDN

When you only need to include Bootstrap's compiled CSS or JS, you can use the Bootstrap CDN.

CSS only

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/bootstrap.min.css">
```

JS, Popper.js, and jQuery

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js"></script>
```

Nadelen van CDN. Wanneer er iets zou wijzigen in de broncode online dan wordt dit meegenomen naar je volledige site. In theorie zouden er dus fouten kunnen optreden. Wij gaan geen gebruik maken van CDN's in deze cursus en zullen de volledige installatie zelf uitvoeren.

1.1.2. INSTALLATIE METHODE 2: PACKAGE MANAGERS

Bootstrap 4 maakt het echter eenvoudiger om een volledige locale installatie uit te voeren op je pc.

Grootste voordeel: wanneer je je site in Bootstrap 4.0 hebt gemaakt dan dien je dit volledige pakket volledig op te laden als site. Je zal dan nooit compatibiliteitsproblemen ondervinden wanneer er latere versies via het CDN netwerk worden verspreid. In deze cursus zullen we dus gebruik maken van de volledige core functionaliteit.

Voor de installatie dienen we gebruik te maken van een package manager. Voorbeelden van package managers zijn npm, meteor,... waarvan Bootstrap u de mogelijkheid geeft om ze te gebruiken. Bootstrap geeft u de keuze tussen meteor en npm

Het is best mogelijk dat je reeds een package manager op je pc hebt geïnstalleerd. Wanneer dit niet het geval is, dan zullen wij gebruiken van npm (node.js) die je kan vinden op de volgende link:
<https://nodejs.org/>

Open uw command prompt en voer de volgende code uit. Tijdens het schrijven van deze cursus bevonden we ons in de versie 4.0 beta 2 van bootstrap. Wanneer je dus een nieuw project zal beginnen dien je op hun website na te zien wat de laatste versie was en deze te gaan gebruiken:

```
$ npm install bootstrap@4.0.0-beta.2
```

OPMERKING: afhankelijk van je pc/mac kan deze installatie serieus verschillen afhankelijk van de versies van node.js (npm) en bootstrap zelf.

Grootste nadeel: Bootstrap installeert soms overbodige mappen die je project niet nodig heeft en zorgen voor een overload.

1.1.3. INSTALLATIE METHODE 3: HANDMATIGE INSTALLATIE

Om het framework volledig vanaf nul te bekijken zullen wij de handmatige installatie van het framework uitvoeren, zodat je als frontend developer weet wat er allemaal nodig is om bootstrap te draaien.

Ga naar: www.getbootstrap.com en klik op de download knop en vervolgens op download source.

Bootstrap

Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

Get started

Download

Currently v4.0.0-beta.2



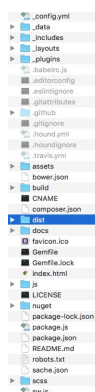
Source files

Compile Bootstrap with your own asset pipeline by downloading our source Sass, JavaScript, and documentation files. This option requires some additional tooling:

- Sass compiler (Libsass or Ruby Sass is supported) for compiling your CSS.
- [Autoprefixer](#) for CSS vendor prefixing

Should you require [build tools](#), they are included for developing Bootstrap and its docs, but they're likely unsuitable for your own purposes.

Download source



Dit geeft je een waslijst aan bestanden die we niet allemaal nodig hebben. Wij zullen gebruik maken de dist folder. Kopieer hieruit enkel de js en css folder naar je project.

1. Maak een nieuw project aan: bootstraptemplate
2. Maak een index.html pagina aan in je project folder

3. Ga naar: <http://getbootstrap.com/docs/4.0/getting-started/introduction/> - hier kan je de startertemplate code kopiëren en plakken in index.html bestand. Let op: bovenstaande link kan verschillen afhankelijk van de versie die je gebruikt.
2. Kopieer de js en css folder van de bootstrap source code nu naar naar je project folder.
3. Download de javascript library JQUERY van <https://www.jquery.com/download> en kies voor "download the compressed, producten jquery 3...". Hernoem vervolgens dit bestand naar jquery-min.js en plak deze in de js folder van je project.
4. De enige uitzondering momenteel is de javascript library POPPER. Zij maken momenteel geen gebruik van gecompileerde bestanden. Hier zullen we de de CDN link moeten gebruiken.
5. We vervangen de CDN links van de BOOTSTRAP.CSS, JQUERY.JS en BOOTSTRAP.JS door onze lokale bestanden.

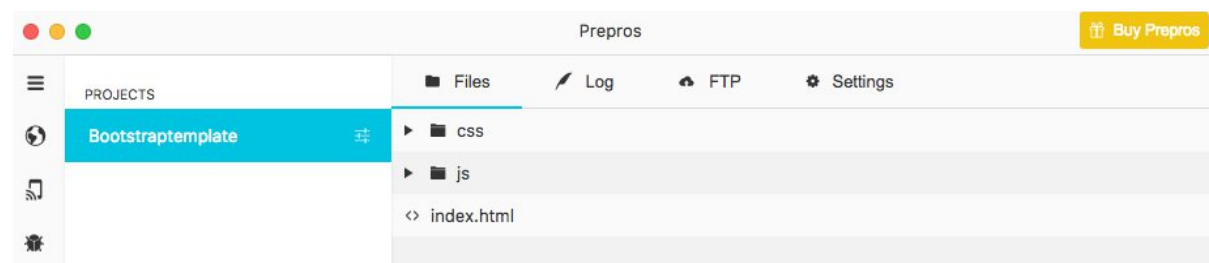
MINIFIED.

We hebben telkens gekozen voor de min versie van de bootstrap, jquery en bootstrap bestanden. De minified versie zorgt ervoor dat alle lijnen code op 1 lijn worden geschreven en de spaties worden overal weggewerkt. Deze code is NIET leesbaar maar zorgt voor een aanzienlijke snelheid bij het laden van een webpagina.

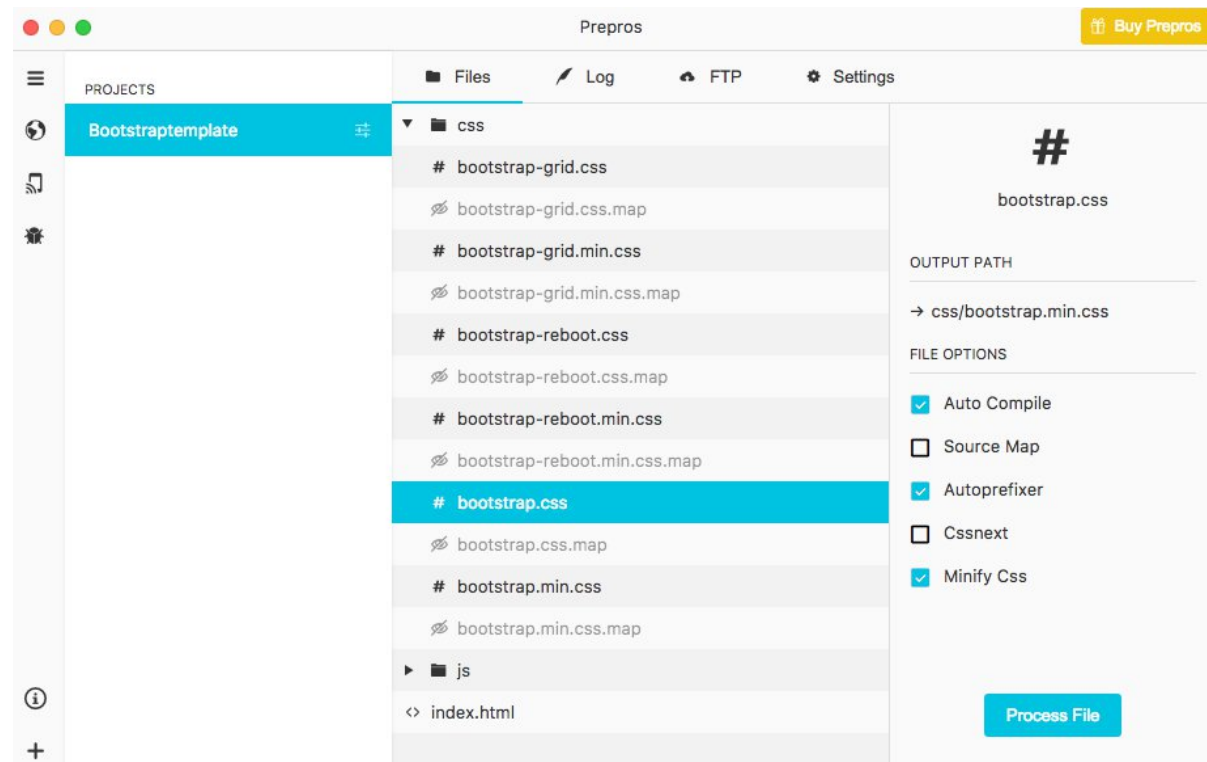
Om minified bestanden toch te kunnen aanpassen maken we gebruik van een tool die we een preprocessor noemen.

Er zijn veel verschillende versies op de markt gaande van gratis preprocessors tot betalende. De applicatie die wij gebruiken is "PREPROS". De keuze is eigenlijk aan jullie als developers wat jullie het meest gebruiksvriendelijk vinden uiteindelijk. PREPROS heeft een gratis versie die wij downloaden en installeren via de link <https://prepros.io/downloads>

6. Start PREPROS op en sleep je project folder in de applicatie



7. We dienen er nu voor de zorgen dat het basis CSS bestand die Bootstrap gebruikt, nl. bootstrap.css telkens weer wordt gecompileerd naar bootstrap.min.css aangezien we die gebruiken in ons html bestand. Je selecteer het basisbestand in de css folder van je project, nl. bootstrap.css maar je wijzigt het output path naar css/bootstrap.min.css. Wanneer wij dus in het basisbestand iets zouden wijzigen dat wordt dit "gepreprocessed" naar de minified versie zoals we die hebben voorzien. Vergeet ook niet de autoprefixer, auto compile en minify css aan te vinken indien dit niet het geval zou zijn. De auto compile zorgt ervoor wanneer je iets wijzigt in het basisbestand dat er gepreprocessed wordt. Dit vervangt eigenlijk de "process file" button die je hier ook ziet staan.



8. SCSS (=SASSY CSS). We tillen CSS3 naar een niveau hoger. We gaan namelijk in de toekomst dynamisch gebruik maken van CSS3. Een webpagina kent echter geen SCSS maar enkel CSS. Daarom moeten we ervoor zorgen dat al onze SCSS bestanden omgezet worden naar onze bootstrap.min.css.

Vanuit je dist folder die je heb gedownload van de site van bootstrap dien je nu ook de volledige folder scss in je project te kopiëren.

9. In de scss folder maak je nu een style.scss bestand aan.

10. In de scss folder kan je het bestand bootstrap.scss terugvinden. Onderaan deze pagina voeg je dan de import toe van het style.scss bestand.

NA BOVENSTAANDE STAPPEN HEB JE EEN VOLLEDIG WERKENDE TEMPLATE VOOR BOOTSTRAP 4.

2. BOOTSTRAP LAYOUT

2.1. BOOTSTRAP GRID SYSTEEM

2.1.1. BOOTSTRAP MOBILE FIRST (12 KOLOMMEN)

Bootstrap werkt met een grid systeem die op **12 kolommen** is gebaseerd EN die **Mobile First** werkt. In een design kunnen we dit makkelijk aanduiden door een laag met 12 kolommen bovenop ons design te plaatsen. Dit doen we zowel op alle layouts die we eventueel ontvangen van een designer. In de meeste gevallen krijgen we een mobiele versie en een desktop versie. In uitzonderlijke gevallen zit er nog een tabletversie bij.

Wanneer we dit toepassen op ons eerste project, nl. Prechu, dan krijg je het volgende resultaat. Bootstrap stelt standaard in zijn framework voor dat extra grote schermen worden weergegeven van een grootte van 1140px met een gutter van 30 px = 1170px.

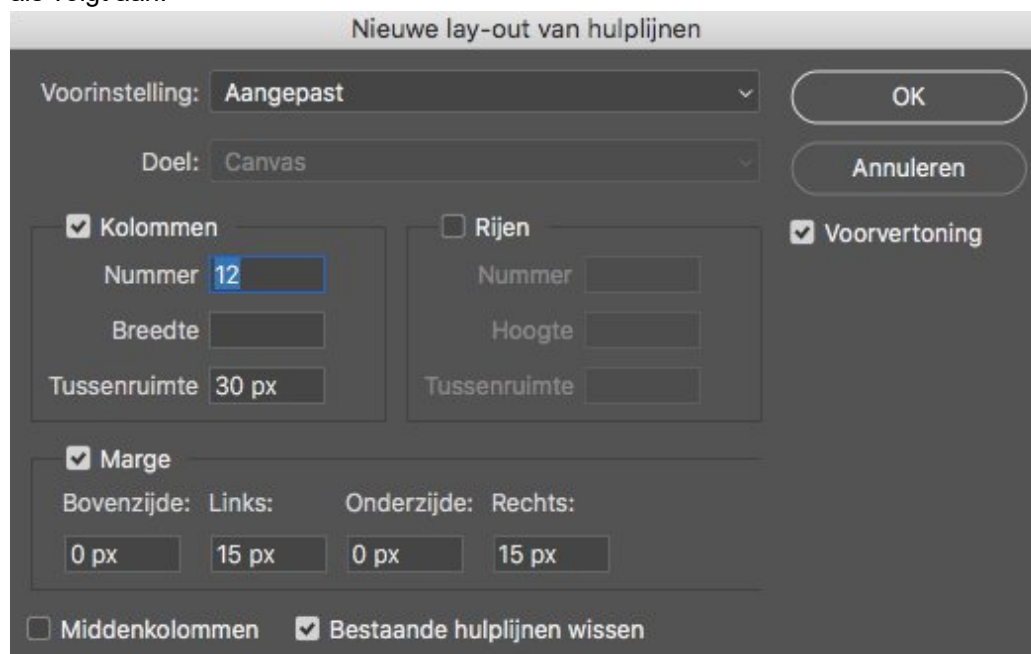
Dit resulteert in 12 kolommen waar de tussenafstand 30px bedraagt. Daarnaast is er ook een gutter voorzien aan de linkerkant van de eerste kolom van 15px en hetzelfde aan de laatste kolom.

DESKTOP

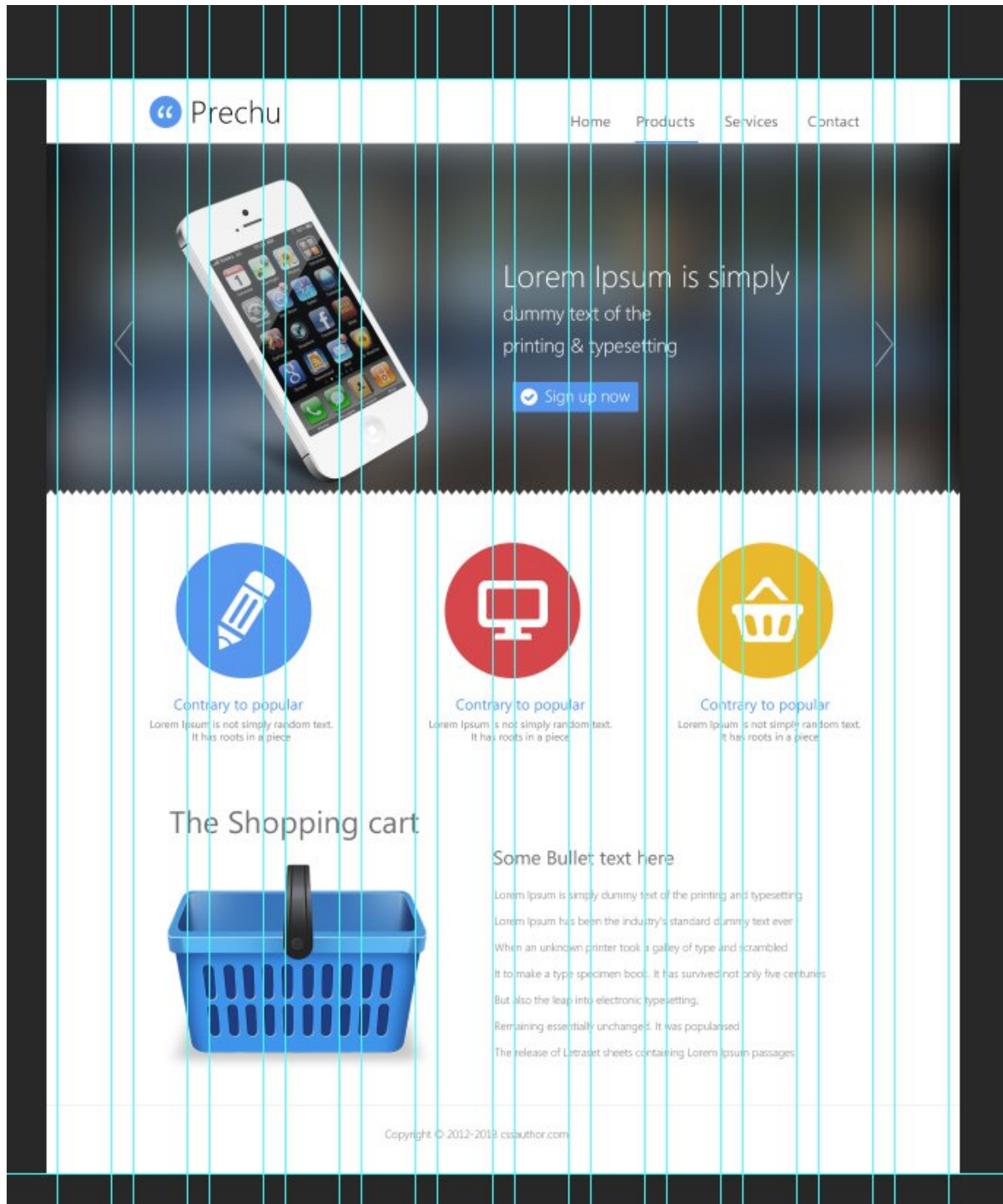
Wanneer we nu ons design (Prechu voorbeeld) openen in photoshop dan dienen we geen rekening te houden met de grootte van het design. Photoshop heeft een handige functie die rekening zal houden met de 12 kolommen die bootstrap voorschrijft.

Deze manier zal rekening houden met de breedte van het design en zal de kolommen gelijk verdelen met de nodige tussenruimte over de volledige breedte van het design. Hier houden we dus geen rekening met de 1140px VASTE containerbreedte.

- Open het Prechu design in photoshop.
- Klik in het menu op **Weergave - Nieuwe layout van hulplijnen maken** en pas de settings als volgt aan:



- Resultaat: je ziet mooi de 12 kolommen met hun tussenruimte



Deze 12 kolommen werken in combinatie met de volgende tag die we in de header vinden van ons index.html bestand:

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

Bovenstaande tag zorgt er voor dat een webpagina wordt weergegeven binnen het toestel die momenteel de webpagina bezoekt, ongeacht dit een smartphone, tablet of desktop is.

viewport = staat letterlijk voor het venster waarin er wordt weergegeven.

content = stelt de breedte van de pagina in volgens de breedte van het toestel die wordt gebruikt.

initial-scale = is het initiële zoomlevel van de pagina.

shrink-to-fit=no = de webpagina mag niet automatisch verkleind worden op tablet of mobiele apparaten

2.1.2.MEDIA-QUERIES - BREAK POINTS (BREEKPUNTEN)

Afhankelijk van de breedte van het scherm die wordt gebruikt om een pagina weer te geven dienen we gebruik te maken van breekpunten in de layout. De standaard breekpunten van bootstrap zijn de volgende:

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

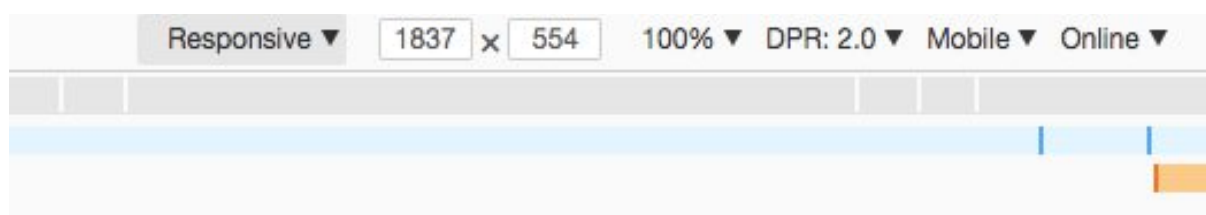
! De algemene regel die wij gaan toepassen in onze cursus is nu de volgende:

< 768px = behandelen we als smartphone weergave

>=768px = behandelen we als tablet weergave

>=992px = behandelen we als desktop weergave.

Deze breekpunten gaan we gebruiken in het inspect element van google chrome. De grijze balk zijn de breekpunten die we gaan gebruiken. Wanneer je je project juist hebt geïnstalleerd met bootstrap dan zijn deze breekpunten te zien in de oranje balk en/of blauwe balk. De blauwe balk toont de maximum breedte van een scherm. De oranje balk staat voor de minimum breedte.



Dit lijkt nu nog allemaal zeer abstract maar wordt zo dadelijk duidelijk in ons eerste voorbeeld.

2.1.3.CONTAINERS

Bootstrap werkt voornamelijk met "classes". De belangrijkste class is de container class. Er zijn 2 soorten containers, nl. **container** en **container-fluid**. Respectievelijk is een container de class die wordt gebruikt voor een vaste breedte van een pagina. De **container-fluid** zorgt voor een volledige schermvulling over de breedte.

Layout - container: standaard 1140px

Layout - container-fluid (uitvulling volledig scherm)

We hebben gezien dat de maximale container breedte standaard 1140px is, die bestaat uit 1110px content en 15px margin aan de rechter- en linkerkant. Deze maximale breedte is het enige dat we zullen aanpassen afhankelijk van ons design die we hebben gekregen.

Open je index.html en vul het volgende in:

```
<div class="container">  
</div>  
  
<div class="container-fluid">  
</div>
```

2.1.4. ROWS

Iedere container, fluid of niet bevat ALTIJD minstens 1 row.

2.1.5. COLUMNS

Iedere row bevat ALTIJD minstens 1 column.

Voorbeeld 1:

Deze container bevat 2 rijen, waarvan de eerste rij 2 kolommen heeft en de tweede rij 3 kolommen. Wanneer we dit op deze manier schrijven dan rekt bootstrap zelf uit hoeveel plaats hij nodig heeft om dit juist te gaan verdelen.

1 of 2		2 of 2	
1 of 3		2 of 3	
		3 of 3	

```
<div class="container">
  <div class="row">
    <div class="col">
      1 of 2
    </div>
    <div class="col">
      2 of 2
    </div>
  </div>
  <div class="row">
    <div class="col">
      1 of 3
    </div>
    <div class="col">
      2 of 3
    </div>
    <div class="col">
      3 of 3
    </div>
  </div>
</div>
```

Voorbeeld 2:

Wanneer we 4 kolommen toch in 1 enkele rij niet naast elkaar willen weergeven maar zoals in het voorbeeld hieronder, dan kunnen we vanaf versie 4 van bootstrap gebruik maken van de class w-100. Dit staat voor een column breedte van 100%. Doet dit weg en je hebt inderdaad 4 columns naast elkaar weergegeven over de beschikbare breedte van het scherm.

Column	Column
Column	Column

```
<div class="container">
  <div class="row">
    <div class="col">Column</div>
    <div class="col">Column</div>
    <div class="w-100"></div>
    <div class="col">Column</div>
    <div class="col">Column</div>
  </div>
</div>
```

Voorbeeld 3:

In dit voorbeeld komen we de eerste keer in contact met ons grid. In de eerste rij zie je dat we gebruik maken van 6 kolommen die uitgerekend worden door bootstrap. De linker- en de rechterkolom worden uitgerekend aan de hand van de beschikbare overgebleven ruimte op het scherm. Enkel dus de middelste kolom heeft een exacte breedte in pixels!

1 of 3	2 of 3 (wider)	3 of 3
1 of 3	2 of 3 (wider)	3 of 3

```
<div class="container">
  <div class="row">
    <div class="col">
      1 of 3
    </div>
    <div class="col-6">
      2 of 3 (wider)
    </div>
    <div class="col">
      3 of 3
    </div>
  </div>
  <div class="row">
    <div class="col">
      1 of 3
    </div>
    <div class="col-5">
      2 of 3 (wider)
    </div>
    <div class="col">
      3 of 3
    </div>
  </div>
</div>
```

Voorbeeld 4:

In dit voorbeeld hebben we te maken met variable breedtes.

De eerste rij heeft naast de normale class row ook de class **justify-content-md-center**. Voorlopig is het genoeg om te weten dat de volledige rij in een container kan worden **gecentreerd**! De linker- en rechterkolom hiervan hebben een vaste breedte van 2 kolommen. De middelste wordt dus uitgevuld met **auto**

1 of 3	Variable width content	3 of 3
1 of 3	Variable width content	3 of 3

```
<div class="container">
  <div class="row justify-content-md-center">
    <div class="col col-lg-2">
      1 of 3
    </div>
    <div class="col-md-auto">
      Variable width content
    </div>
    <div class="col col-lg-2">
      3 of 3
    </div>
  </div>
  <div class="row">
    <div class="col">
      1 of 3
    </div>
    <div class="col-md-auto">
      Variable width content
    </div>
    <div class="col col-lg-2">
      3 of 3
    </div>
  </div>
</div>
```

We kunnen ze nog enige tijd doorgaan. Bootstrap kent enorm veel classes. Een CHEAT SHEET kan hierbij helpen!

2.2. CHEAT SHEET BOOTSTRAP CLASSES

Om je een idee te geven wat de omvang is van bootstrap 4 en zijn classes, surf dan naar onderstaande link:

<https://bootstrapcreative.com/resources/bootstrap-4-css-classes-index/>

Met dank aan bootstrapcreative.com. Er zijn nog gekken in de wereld.

We zullen zeer veel van deze classes gebruiken. De beste handleiding voor bootstrap staat natuurlijk hiervoor op getbootstrap.com met alle code snippets die je maar kan denken. De docent zal u begeleiden om wegwijs te raken in deze handleiding online.

Vanaf hier gaan we verder met bootstrap en maken we ons praktijkvoorbeeld Prechu.

Alerts

- alert-primary
- alert-secondary
- alert-success
- alert-info
- alert-warning
- alert-danger
- alert-light
- alert-dark
- alert-link
- alert-dismissible
- alert-heading

Badges

- badge
- badge-pill
- badge-primary
- badge-secondary
- badge-success
- badge-info
- badge-warning
- badge-danger
- badge-light
- badge-dark

Breadcrumbs

- breadcrumb

Buttons

- btn-primary
- btn-secondary
- btn-success
- btn-info
- btn-warning
- btn-danger
- btn-light
- btn-dark
- btn-link
- btn-outline-primary
- btn-outline-secondary
- btn-outline-success
- btn-outline-info
- btn-outline-warning
- btn-outline-danger
- btn-outline-light
- btn-outline-dark

Button Groups

- btn-group
- btn-group-lg
- btn-group-sm
- btn-group-vertical
- btn-group (nested)
- btn-toolbar

Button Modifiers

- btn-lg
- btn-sm
- btn-block
- active button
- disabled button
- checkbox as button
- radio as button

Cards

- card
- card-body
- card-title
- card-subtitle
- card-text
- card-link
- card-img-top
- middle image
- card-img-bottom
- card-img-overlay
- list-group
- card-header
- h*.card-header
- card-footer
- card-group
- card-deck
- card-columns
- card bg-... text-...

Carousel

- carousel slide
- carousel-fade
- carousel-indicators
- carousel-caption

Collapse

- collapse
- accordion

Dropdowns

- dropdown
- dropdown (split)
- dropup
- dropup (split)
- dropright
- dropleft
- dropdown-header
- dropdown-item-text
- dropdown-divider
- dropdown-item disabled
- dropdown-menu-right

Forms

- form-group
- form-inline
- form using the grid
- form-control
- form-control-lg
- form-control-sm
- form-control-file
- form-control-plaintext
- form-control-range
- form-check
- form-check-inline
- disabled items
- readonly

Form Input Groups

- input-group
- input-group-prepend
- input-group-append
- input-group-sm
- input-group-lg
- checkbox
- radio
- dropdown
- segmented buttons

Custom Forms

- custom-checkbox
- custom-radio
- custom-switch
- custom-select
- custom-file
- custom-range

Grid

- container
- container-fluid
- row
- col-# (<576px)
- col-sm-# (≥576px)
- col-md-# (≥768px)
- col-lg-# (≥992px)
- col-xl-# (≥1200px)
- col
- col-*
- no-gutters
- offset-*-#
- order-#
- nested columns

Images

- img-fluid
- img-thumbnail

Jumbotron

- jumbotron
- jumbotron-fluid

List Group

- list-group
- list-group-item active
- list-group-item disabled
- list-group-item-action
- list-group-item-primary
- list-group-item-secondary
- list-group-item-success
- list-group-item-info
- list-group-item-warning
- list-group-item-danger
- list-group-item-light
- list-group-item-dark
- list-group with badges
- list-group with d-flex
- list-group-horizontal

Media Objects

- media
- nested media
- d-flex align-self-start
- d-flex align-self-center
- d-flex align-self-end
- right aligned media

Modal

- modal
- modal-dialog-centered
- modal-dialog-scrollable
- modal fade
- modal-lg
- modal-sm
- modal-xl

Navs

- ul.nav
- nav.nav
- nav justify-content-*
- nav flex-column
- nav-tabs
- nav-pills
- nav-fill
- nav-justified
- nav with flex utils
- nav-tabs with dropdown
- nav-pills with dropdown

Navbar

- navbar
- navbar-brand
- navbar with form
- navbar-text
- navbar-dark bg-dark
- navbar-light
- navbar fixed-top
- navbar fixed-bottom
- navbar sticky-top
- collapse navbar-collapse
- navbar-toggler
- navbar-expand-*

Pagination

- pagination
- page-item disabled
- page-item active
- pagination-lg
- pagination-sm

Popover

- popovers
- dismissible popover

Progress

- progress
- progress-bar
- progress-bar with label
- progress-bar with height
- progress-bar bg-*
- multiple progress-bar
- progress-bar-striped
- progress-bar-striped bg-*
- progress-bar-animated

Scrollspy

- data-spy

Spinner

- spinner-border
- spinner-border text-*
- spinner-border-sm
- spinner-grow
- spinner-grow text-*
- spinner-grow-sm

Tables

- table
- thead-light
- thead-dark
- table-striped
- table-bordered
- table-borderless
- table-hover
- table-sm
- table-*-responsive
- table-reflow
- table-active
- table-primary
- table-secondary
- table-success
- table-info
- table-warning
- table-danger
- table-light
- table-dark

Toasts

- toast

Tooltips

- tooltip

Typography

- display-# (1-4)
- lead
- blockquote
- blockquote-footer
- blockquote-reverse
- list-unstyled
- list-inline
- dl-horizontal

Utility: Borders

- border
- border-*-0
- border-primary
- border-secondary
- border-success
- border-danger
- border-warning
- border-info
- border-light
- border-dark
- border-white
- rounded
- rounded-sm
- rounded-lg
- rounded-circle
- rounded-0
- rounded-*

Utility: Colors

- bg-primary
- bg-secondary
- bg-success
- bg-info
- bg-warning
- bg-danger
- bg-light
- bg-dark
- bg-white
- text-primary
- text-secondary
- text-success
- text-info
- text-warning
- text-danger
- text-light
- text-dark
- text-white

Utility: Display

- d-*-block
- d-*-flex
- d-*-inline
- d-*-inline-block
- d-*-inline-flex
- d-*-none
- d-*-table
- d-*-table-cell
- d-print-...

Utility: Flexbox

- flex-*-column
- flex-*-column-reverse
- flex-*-row
- flex-*-row-reverse
- flex-*-nowrap
- flex-*-wrap
- flex-*-wrap-reverse
- flex-fill
- flex-*-grow-1
- flex-*-grow-0
- flex-*-shrink-1
- flex-*-shrink-0
- justify-content-*-start
- justify-content-*-end
- justify-content-*-center
- justify-content-*-between
- justify-content-*-around
- align-content-*-start
- align-content-*-end
- align-content-*-center
- align-content-*-around
- align-content-*-stretch
- align-items-*-start
- align-items-*-end
- align-items-*-center
- align-items-*-baseline
- align-items-*-stretch
- align-self-*-start
- align-self-*-end
- align-self-*-center
- align-self-*-baseline
- align-self-*-stretch
- order-*-#

Utility: Misc

- close
- embed-responsive
- shadow-none
- shadow-sm
- shadow
- shadow-lg
- invisible
- visible
- sr-only
- sr-only-focusable
- stretched-link

Utility: Positioning

- align-*
- clearfix
- fixed-top
- fixed-bottom
- sticky-top
- float-*-left
- float-*-right
- float-*-none

Utility: Sizing

- w-100
- h-100
- mw-100
- mh-100

Utility: Spacing

- m-1 / m-* -#
- mt-1 / mt-* -#
- mr-1 / mr-* -#
- mb-1 / mb-* -#
- ml-1 / ml-* -#
- mx-1 / mx-* -#
- my-1 / my-* -#
- m-n1 / m-* -n#
- p-1 / p-* -#
- pt-1 / pt-* -#
- pr-1 / pr-* -#
- pb-1 / pb-* -#
- pl-1 / pl-* -#
- px-1 / px-* -#
- py-1 / py-* -#

Utility: Text

- font-weight-bold
- font-weight-bolder
- font-weight-light
- font-weight-lighter
- font-weight-normal
- text-decoration-none
- font-italic
- text-monospace
- text-justify
- text-nowrap
- text-*-left
- text-*-right
- text-*-center
- text-lowercase
- text-uppercase
- text-capitalize
- text-truncate
- text-body
- text-black-50
- text-white-50
- text-muted
- text-hide
- text-break

3. SASS

3.1. WAT IS SASS

SASS (Syntactically Awesome Stylesheets of sassy css) is een CSS-pre-processor waarmee u variabelen, wiskundige bewerkingen, mixins, loops, functies, import en andere interessante functies kunt gebruiken die CSS veel krachtiger maken. In sommige opzichten zou je SASS kunnen beschouwen als een stijltaaluitbreidingstaal omdat het de standaard CSS-kenmerken uitbreidt door de voordelen van een eenvoudige programmeertaal te introduceren. Dus SASS zal uw code compileren en de CSS-output genereren die een browser kan begrijpen.

Als je eenmaal CSS met SASS hebt geleerd, schrijf je geen CSS meer vanaf nul.

Als u CSS kent, kent u SASS. SASS wordt geleverd met twee verschillende syntaxis: SASS zelf en SCSS, de meest gebruikte. SCSS-syntaxis is CSS-compatibel, dus u hoeft alleen uw .css-bestand te hernoemen naar .scss et voilà! Je eerste SCSS-bestand is zo gemaakt.

steeds opnieuw herhalen, maar met SASS kunt u deze waarden als variabelen opslaan:

```
$blauw: # 004BB4;
$raleway-font: 'Raleway', 'Arial', 'Helvetica', sans-serif;
$nunito-font: 'Nunito', 'Arial', 'Helvetica', sans-serif;
Nadat u de variabelen hebt gemaakt, kunt u ze gebruiken waar u maar wil:
h1 {
  font-family : $raleway-font;
  color : $blauw;
}
p {
  font-family : $nunito-font;
  background : $blauw;
  padding : 6 px;
}
```

3.1. NESTEN VAN CSS

Het nesten van css heeft enkele voordelen:

- Gemakkelijk leesbaar.
- Voorkomt dat u selectors meerdere keren moet herschrijven
- Betere structuur van code dankzij de visuele hiërarchie.
- Meer onderhoudbare code.

```
.navbar {  
  font-family : $ raleway-font;  
  color : $ blauw;  
  li {  
    a {  
      font-size : 1,5 rem;  
      span {  
        weight : 600;  
      }  
    }  
  }  
}
```

3.1. MIXINS

Het gebruik van variabelen is geweldig, maar wat als je blokken code meer dan eens in je stylesheet hebt? Dat is wanneer mixins in het spel komen. Mixins zijn soortgelijke functies in andere programmeertalen. Ze retourneren een waarde of een reeks waarden en kunnen parameters bevatten, waaronder standaardwaarden. Merk op dat SASS ook functies heeft, dus verwar een mixin niet met een functie.

Laten we een voorbeeld bekijken. Stel je voor dat je deze mixin hebt:

```
@mixin set-font ($family: 'Raleway ', $weight: 400, $style: normal)  
{  
  font-family: $family, 'Arial', 'Helvetica', sans-serif;  
  font : $style ;  
  font-weight : $weight;  
}
```

Nu u uw mixin heeft gedefinieerd, kunt u deze gebruiken waar u maar wil. Let op: omdat u standaardwaarden hebt opgegeven, hoeft u geen parameter door te geven:

```
h1 {  
  @include set-font;  
  color : $blauw;  
}
```

Bovenstaande wordt uiteindelijk naar css gecompileerd als volgt:


```
h1 {  
  font-family: 'Raleway ', 'Arial', 'Helvetica', sans-serif;  
  font-family : normal ;  
  weight : 400;  
  color : #004BB4;  
}
```

Als u de standaard mixin-waarden wilt bijwerken, hoeft u alleen de parameters door te geven binnen de @include-aanroep.

```
h1.callout {  
  @inclusief set-font ('Raleway ', 600);  
  color : $blauw;  
}
```

VERDEEL JE CODE MET IMPORT! Met import zijn de variabelen globaal bereikbaar. In de nieuwe versies die er aan zitten te komen wordt aangeraden @use te gebruiken ipv @import.

```
@import "mijn aangepaste scss";
```