

Development 1 : Labo CALC

studiegebied **HWB**

bachelor in het **toegepaste Informatica**

campus **Kortrijk**

academiejaar **2023-2024**

CSS calculated values

De CSS `calc()` functie kan gebruikt worden om simpele berekeningen (calculations) in je CSS regels uit te voeren. Hiermee kan je de value van een CSS property gaan berekenen i.p. v. een hardcoded waarde mee te geven.

In plaats van bijvoorbeeld een statische waarde in pixels mee te geven als width van een element, kan met behulp van `calc()` de width van dit element gespecificeerd worden als een berekend resultaat van 2 of meerdere waardes.

In de `calc()` functie kan je gebruik maken van de mathematische expressies: `+`, `-`, `*`, `/`

Syntax

```
.item{  
  width: calc(100% - 60px);  
}
```

Het is perfect mogelijk om verschillende maateenheden te gaan gebruiken in de `calc()` functie. Zo kan je bijvoorbeeld (zie voorbeeld hierboven) pixels aftrekken van procenten...

De standaard volgorde van berekeningen wordt toegepast:

- Alle berekeningen worden uitgevoerd van links naar rechts.
- Vermenigvuldigen en delen hebben voorrang op optellen en aftrekken.
- Berekeningen tussen haakjes worden eerst uitgevoerd.

Nesting

`calc()` functies kunnen genest worden. Deze geneste `calc()` functies worden op dezelfde manier uitgevoerd als een berekening tussen haakjes.

Voorbeeld:

```
.item {  
  width: calc( 100% / calc(100px * 2) );  
}
```

In uiteraard hetzelfde als:

```
.item {  
  width: calc( 100% / (100px * 2) );  
}
```

Browser support van de calc() functie

<https://caniuse.com/#feat=calc>

Voor browsers die de `calc()` functie niet ondersteunen wordt de volledige CSS-regel (property: value) genegeerd. Hierdoor kan op een eenvoudige manier als fallback een statische waarde worden gedefinieerd. Browsers die de `calc()` functie ondersteunen zullen deze statische waarde dan meteen gaan overschrijven.

Zie voorbeeld:

```
.item {  
    width: 90%; /* Fallback for older browsers */  
    width: calc(100% - 50px);  
}
```

Opdrachten

Opdracht 1

Gegeven:

index.html

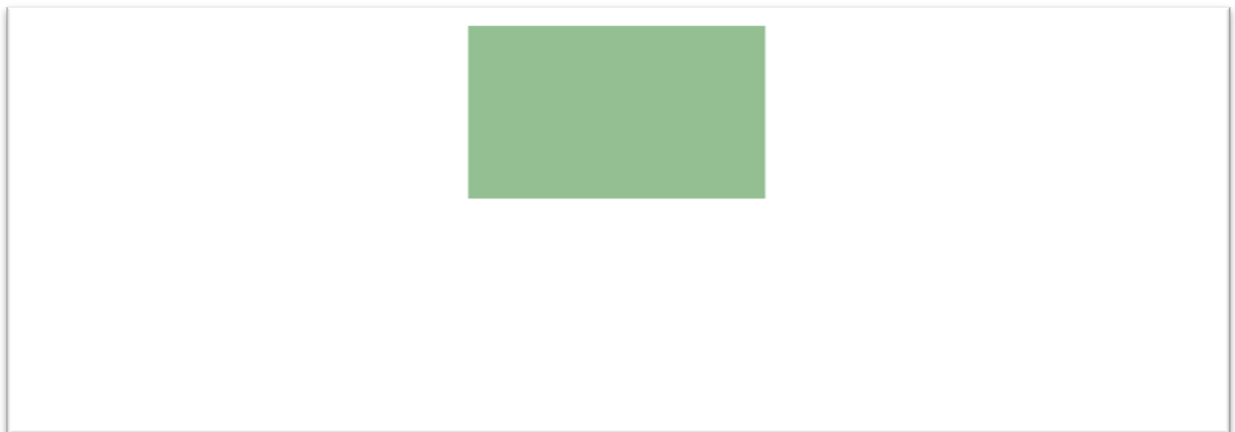
```
<body>
  <div id="greenBlock"></div>
</body>
```

style.css gekoppeld aan index.html

```
#greenBlock {
  width: 500px;
  height: 300px;
  background-color: darkseagreen;

  /* centreer horizontaal */
  margin: 0px auto;
}
```

Resultaat in browser:



Gevraagd:

Gebruik niet langer de css property `margin` om het groene blok gecentreerd te krijgen, maar bereken waar het groene blok absoluut gepositioneerd moet worden ten opzichte van de linker zijkant (`left`).

Opdracht 2

Gegeven

index.html

```
<body>
  <div class="col column-1-7"></div>
  <div class="col column-2-7"></div>
  <div class="col column-3-7"></div>
  <div class="col column-1-7"></div>
</body>
```

style.css gekoppeld aan index.html

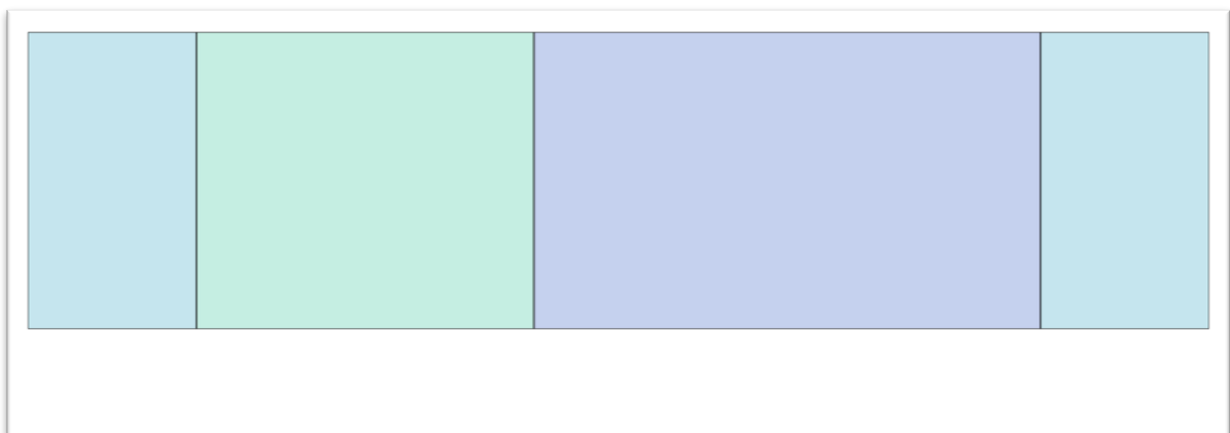
```
body {
  margin: 20px;
  padding: 0;
}

* {
  box-sizing: border-box;
}

.column-1-7 {
  width: 14.2857%;
  background-color: #c2e3ec;
}
.column-2-7 {
  width: 28.5714%;
  background-color: #c2ece0;
}
.column-3-7 {
  width: 42.8571%;
  background-color: #c2ceec;
}

.col {
  float: left;
  height: 500px;
  border: 1px solid black;
}
```

Resultaat in browser:



Gevraagd:

Met behulp van de CSS-classes `.column-x-7` kan een parent element opgedeeld worden in 7 kolommen. De som van de 'x' in de gebruikte CSS-class bij alle childs moet 7 zijn. Op die manier wordt de parent volledig opgevuld. Dit kan bijvoorbeeld zijn:

- 7x `<div>` met class `.column-1-7`
- Of zoals in bovenstaande voorbeeld (samen = 7):
 - `.column-1-7`
 - `.column-2-7`
 - `.column-3-7`
 - `.column-1-7`

Hoewel de width van deze CSS-classes al vrij correct is bepaald (tot 4 cijfers na de komma), is het niet helemaal correct. $100\% / 7 = 14.2857142857....$

Herschrijf de 3 CSS-classes (`.column-1-7`, `.column-2-7`, `.column-3-7`) zodat de width via de `calc()` functie wordt bepaald.

Opdracht 3

Gegeven

index.html

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

style.css gekoppeld aan index.html

```
img {  
  position: absolute;  
}  
  
#arch {  
  top: 100px;  
  left: 100px;  
}  
  
#tech {  
  bottom: 100px;  
  right: 100px;  
}  
  
#animals {  
  top: 100px;  
  right: 100px;  
}  
  
#nature {  
  left: 100px;  
  bottom: 100px;  
}
```

Resultaat in browser:



Gevraagd

Herschrijf de CSS-regels waardoor je exact hetzelfde resultaat verkrijgt, maar dit keer zonder gebruik te maken van de CSS-properties: `right` en `bottom`

Iedere ID-selector: `#arch`, `#tech`, `#animals` & `#nature` bevat dus enkel 2 CSS-properties: `top` en `left`

Tips:

- hou rekening met de afmetingen van de 4 afbeeldingen
- De totale hoogte en breedte van je scherm === 100%