

# HoGent

BEDRIJF  
EN  
ORGANISATIE

## Webapplicaties 1

Layouting III: Flexbox

HoGent

## Inhoud

---

- ▶ Inleiding
- ▶ Flex container
  - flex-direction
  - flex-wrap
  - justify-content
  - align-items
  - align-content
- ▶ Flex items
  - order
  - flex-grow & flex-shrink
  - flex-basis
- ▶ Absolute & relative flex
- ▶ Flexbox en margin: auto;
- ▶ Flexbox in column mode

HoGent

2

## Inleiding

- ▶ Flexbox is een nieuwe manier in CSS3 om **delen van je webapplicatie** eenvoudig te layouten
- ▶ Lost moeilijkheden zoals verticaal centereren in “gewone CSS” op
- ▶ Basisidee: elementen positioneren langs **assen**
  - Er is **geen horizontaal of verticaal**, enkel de richting van de assen

HoGent

3

## Browserondersteuning

- ▶ Zie [caniuse.com](http://caniuse.com)

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			49						
			59					4.4	
	14	54	60			10.2		4.4.4	
11	15	55	61	10.1	47	10.3	all	56	61
	16	56	62	11	48	11			
		57	63	TP	49				
		58	64						

HoGent

4

## Flex container

- ▶ Element wiens kinderen op *flexibele* wijze getoond zullen worden
  - De kinderen zijn *flex items*
- ▶ *display: flex* in CSS

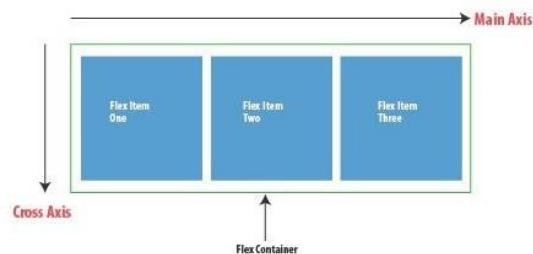


HoGent

5

## Flex containers


- ▶ Flex containers hebben een **main axis** en een **cross axis**
  - Standaard gaat cross axis van links naar rechts, cross axis van boven naar onder
  - Wordt aangepast met *flex-direction* property
- ▶ De container wordt opgevuld langs de main axis





HoGent


## Flex containers: flex-direction

- flex-direction bepaalt de richting van de **main axis**

- flex-direction: row; 

- flex-direction: column; 

- flex-direction: row-reverse; 

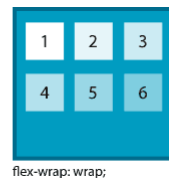
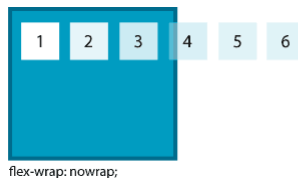
- flex-direction: column-reverse; 

HoGent

## Flex containers: flex-wrap

- Wat als er de flex container vol geraakt? **Bepaalt door flex-wrap.**

- flex-wrap: nowrap;
- flex-wrap: wrap;
- flex-wrap: wrap-reverse;

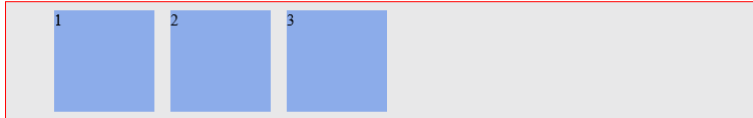


HoGent

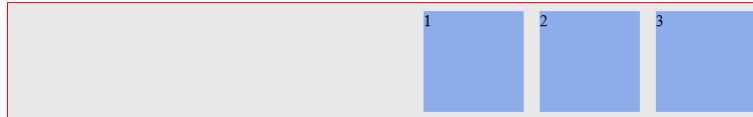
## Flex containers: justify-content

► Items positioneren langs main axis met **justify-content**

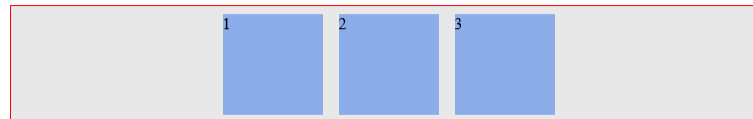
- justify-content: flex-start; (default)



- justify-content: flex-end;



- justify-content: center;



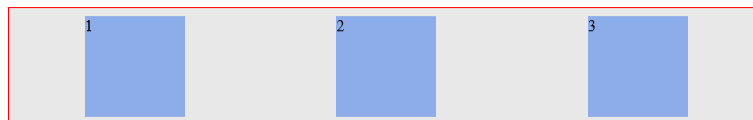
HoGent

9

## Flex containers: justify-content

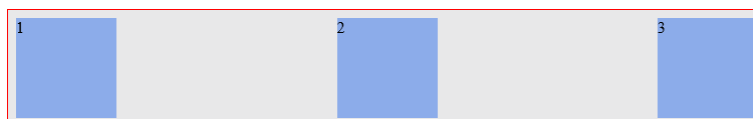
- justify-content: space-around;

- **Rond** elk item evenveel witruimte



- justify-content: space-between

- **Tussen** elk item evenveel witruimte



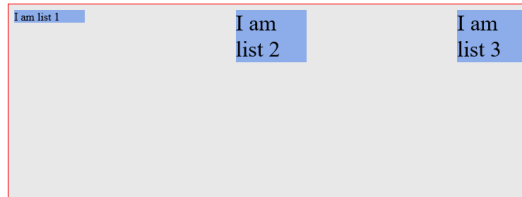
HoGent

10

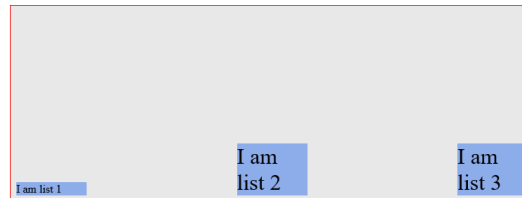
## Flex containers: align-items

### ► Items positioneren langs cross axis met **align-items**

- align-items: flex-start;



- align-items: flex-end;

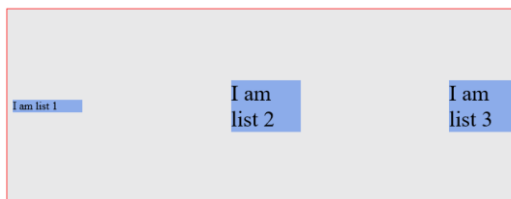


HoGent

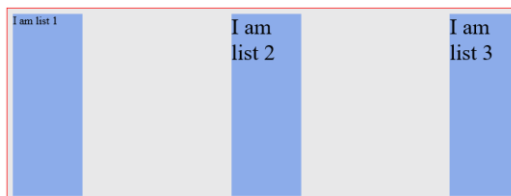
11

## Flex containers: align-items

- align-items: center;



- align-items: stretch;

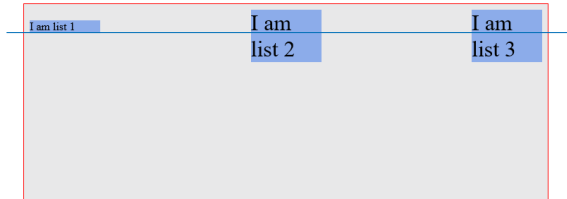


HoGent

12

## Flex containers: align-items

- align-items: baseline;
- aligneert items volgens “onderkant” tekst.



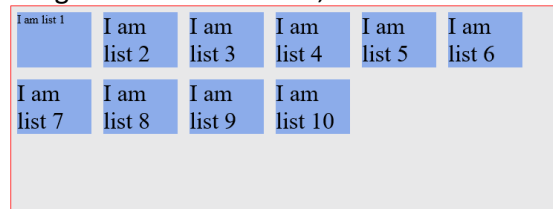
HoGent

13

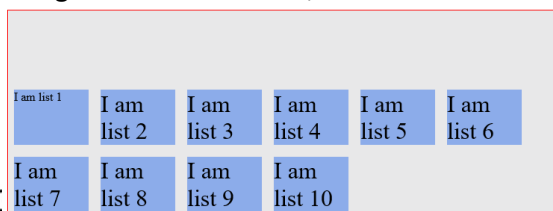
## Flex containers: align-content

- **Volledige inhoud container** positioneren langs cross axis met **align-content**

- align-content: flex-start;



- align-content: flex-end;

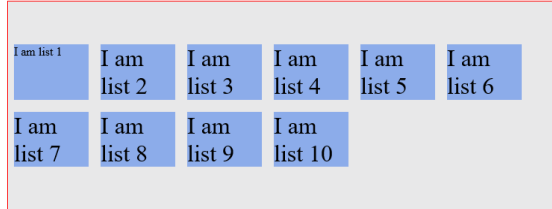


HoC

14

## Flex containers: align-content

- align-content: center;



- align-content: stretch;

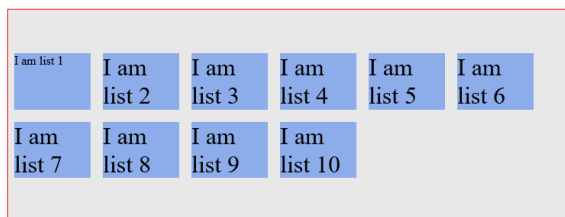


HoGent

15

## Flex containers: align-content

- align-content: baseline;



HoGent

16

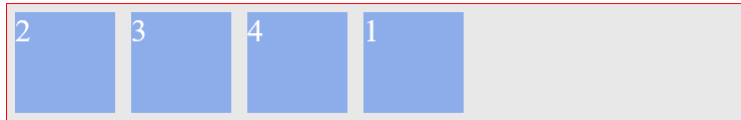


## Flex items: order

- ▶ Volgorde waarin de items getoond worden wijzigen, zonder HTML aan te passen
  - Standaardwaarde is 0
  - Items worden geordend van klein naar groot

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li:nth-child(1) {
  order: 1;
}
```



HoGent

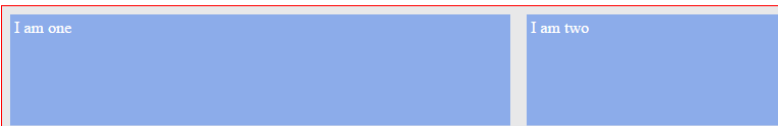
17

## Flex items: flex-grow & flex-shrink

- ▶ *flex-grow* en *flex-shrink* bepalen **hoeveel een item mag groeien/verkleinen** als er extra plaats is in de container
- ▶ Waarde: getal
  - 0: niet groeien
  - Positief: groei in verhouding met andere items

```
li:nth-child(1) {
  flex-grow: 2;
}
```

```
li:nth-child(2) {
  flex-grow: 1;
}
```



HoGent

18

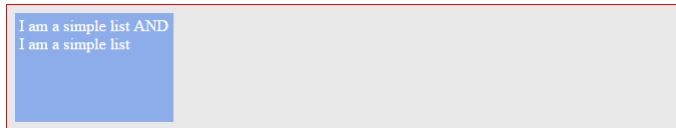
## Flex items: flex-basis

---

- *flex-basis* bepaalt de initiele grootte van een item voordat *flex-grow* en *flex-shrink* worden toegepast.
  - Normaal



- *flex-basis: 150px;*



HoGent

19

## Flex items: flex

---

- De *flex* property laat je toe *flex-grow*, *flex-shrink* en *flex-basis* in een keer te definiëren
  - `flex-grow: 2;`  
`flex-shrink: 1;`  
`flex-basis: auto;`

} `flex: 2 1 auto;`

HoGent

20

## Absolute & relative flex

- ▶ Absolute flex items: ingenomen ruimte **enkel bepaald door Flexbox**
  - flex: 1 1 0;
  - item mag groeien, mag verkleinen, en er wordt **geen ruimte bepaald op voorhand**

This is just some random text to buttress the point been explained. Some more random text to buttress the point being explained.

This is just a shorter random text.

- ▶ Relative flex items: ingenomen ruimte **enkel bepaald door grootte inhoud**
  - flex: 1 1 auto;
  - item mag groeien, mag verkleinen, maar **ruimte wordt eerst automatisch bepaald door inhoud**

This is just some random text to buttress the point been explained. Some more random text to buttress the point being explained.

This is just a shorter random text.

HoGent

21

## Flexbox en margin: auto;

- ▶ margin: auto instellen op een item zal vrij ruimte “verplaatsen”

```
<ul>
  <li>Branding</li>
  <li>Home</li>
  <li>Services</li>
  <li>About</li>
  <li>Contact</li>
</ul>
```

```
ul {
  display: flex;
}
li {
  flex: 0 0 auto;
}
```

Branding Home Services About Contact

HoGent

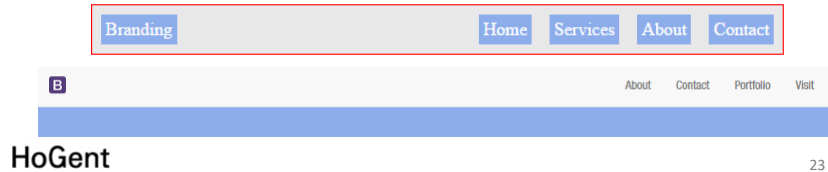
22

## Flexbox en margin: auto;

- Rechtermargin instellen zorgt dat daar de vrije ruimte geplaatst wordt

```
<ul>
  <li>Branding</li>
  <li>Home</li>
  <li>Services</li>
  <li>About</li>
  <li>Contact</li>
</ul>
```

```
ul {
  display: flex;
}
li {
  flex: 0 0 auto;
}
li:nth-child(1) {
  margin-right: auto;
}
```

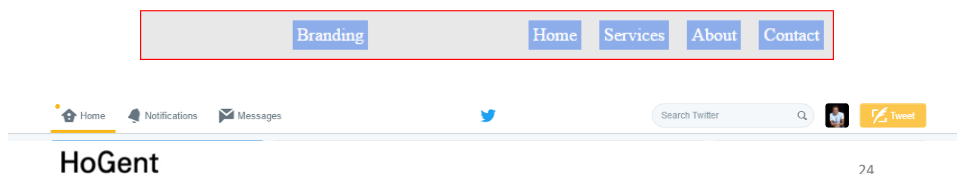


## Flexbox en margin: auto;

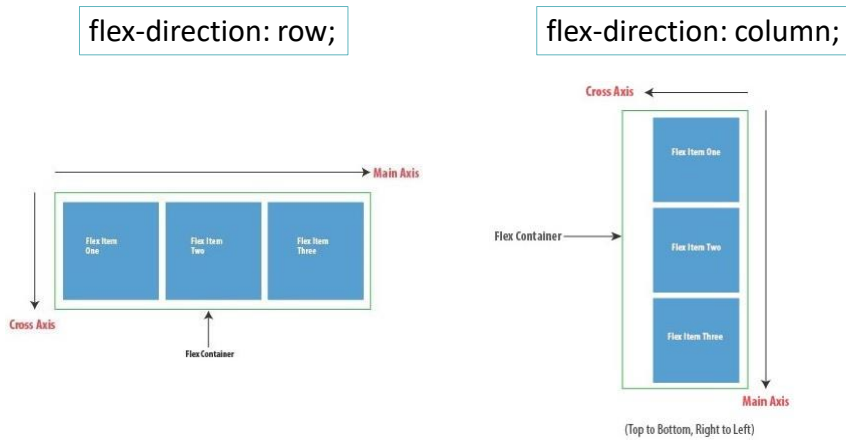
- Beide margins instellen zorgt dat de vrije ruimte langs beide kanten verspreid wordt

```
<ul>
  <li>Branding</li>
  <li>Home</li>
  <li>Services</li>
  <li>About</li>
  <li>Contact</li>
</ul>
```

```
ul {
  display: flex;
}
li {
  flex: 0 0 auto;
}
li:nth-child(1) {
  margin-left: auto;
  margin-right: auto;
}
```



## Flexbox in column-mode

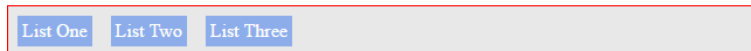


HoGent

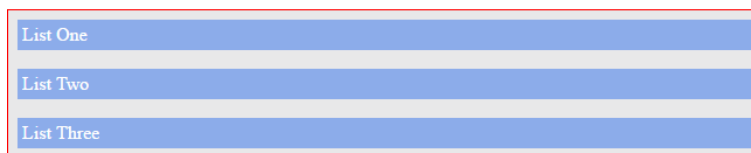
25

## Flexbox in column-mode

- `flex-direction: row;`



- `flex-direction: column;`



HoGent

26

## Flexbox in column-mode

---

- ▶ flex-basis stelt ruimte in **langs main axis**
  - flex-direction: column;
  - flex-basis: 100px;



HoGent

27

## Flexbox in column-mode

---

- ▶ Items horizontaal centreren
  - justify-content? Werkt langs **main axis**
  - align-items? Werkt langs **cross axis**

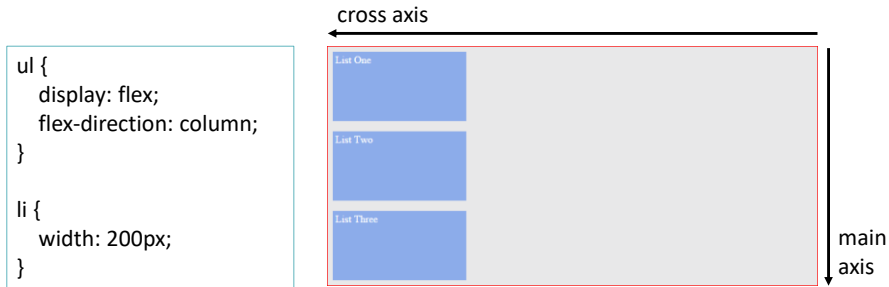


HoGent

28

## Flexbox in column-mode

- ▶ Items horizontaal centreren
  - justify-content? Werkt langs **main axis**
  - align-items? Werkt langs **cross axis**



HoGent

29

## Flexbox in column-mode

- ▶ Items horizontaal centreren
  - justify-content? Werkt langs **main axis**
  - align-items? Werkt langs **cross axis**

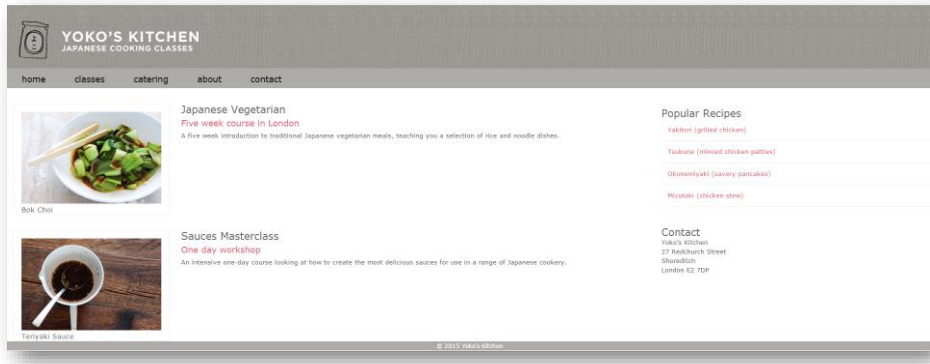


HoGent

30

## Voorbeeld

- Hier werd flex layout gebruikt voor de opmaak van de yoko's kitchen



HoGent

31

## Referenties

- Understanding Flexbox: Everything you need to know,  
<https://medium.freecodecamp.org/understanding-flexbox-everything-you-need-to-know-b4013d4dc9af>
- A guide to Flexbox,  
<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- Responsive design of the future with Flexbox,  
<http://blog.teamtreehouse.com/responsive-design-of-the-future-with-flexbox>
- CSS Flexible Box Layout,  
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout)

HoGent

32