

# Advanced HTML: Flexbox

Pluralsight:

<https://app.pluralsight.com/library/courses/css-flexbox-fundamentals-2319>

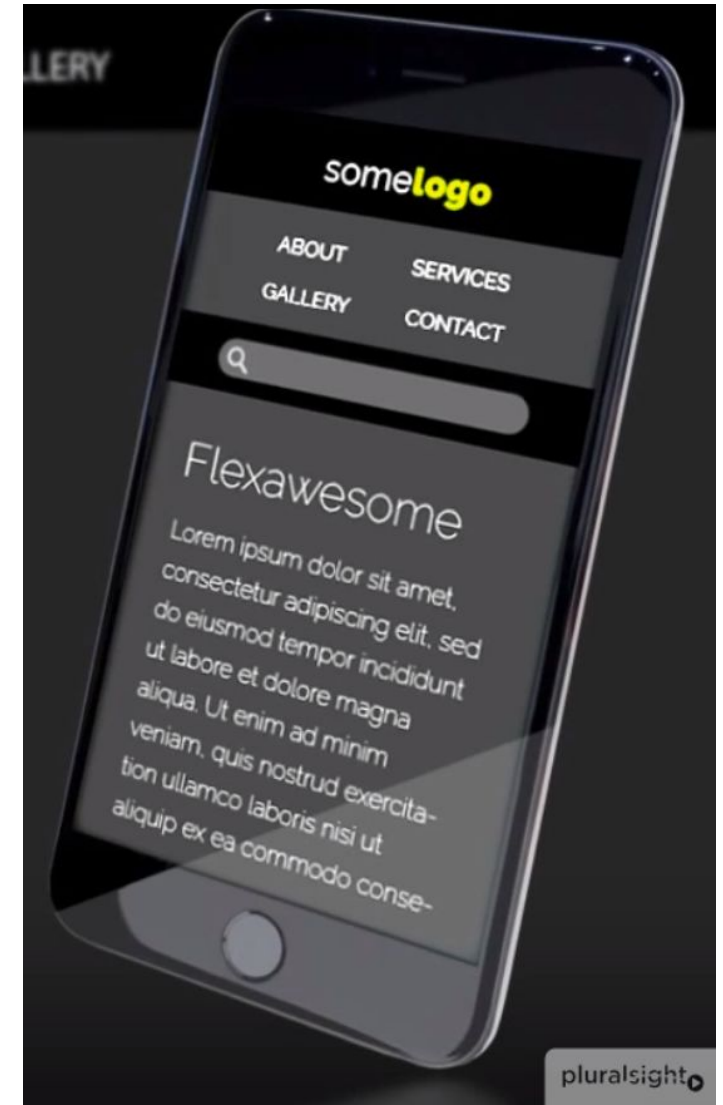


# Flexbox

- Nieuwe manier om layouts te maken
- Geen floats meer!
- Geen layout-hacks meer! (clearfix e.d.)

Officiële documentatie:

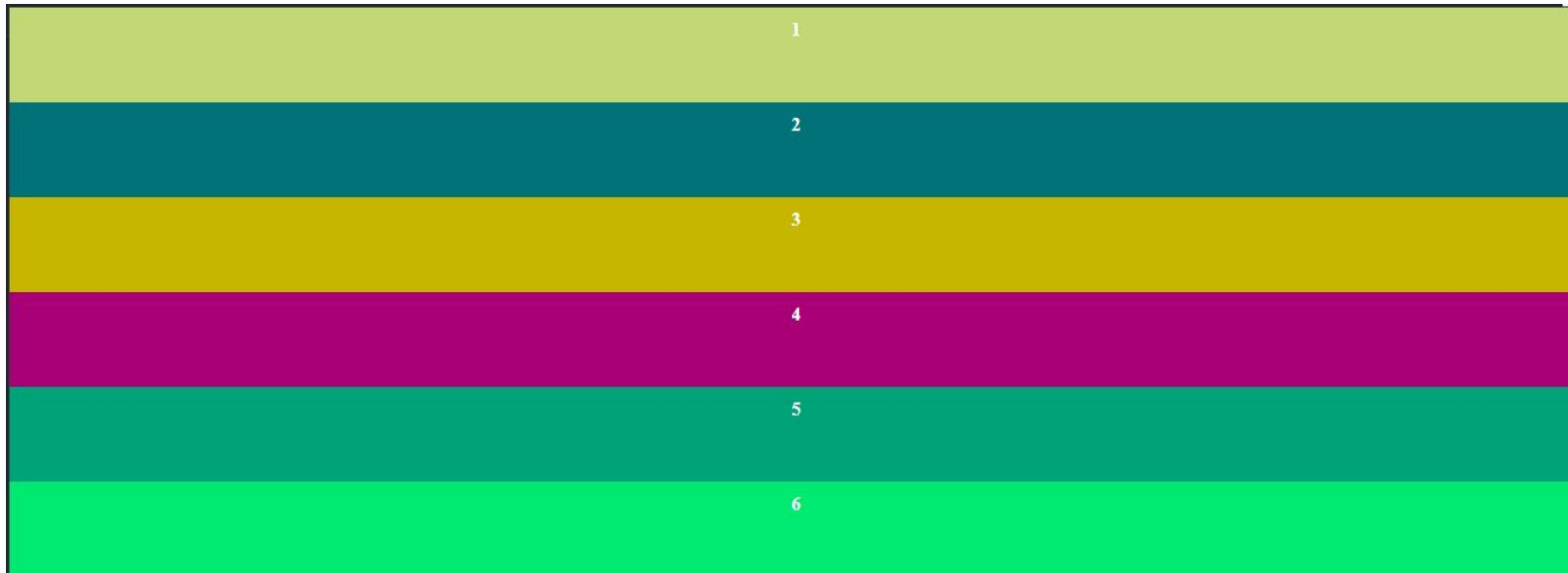
[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout/Basic\\_Concepts\\_of\\_Flexbox](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox)



# Flexbox playground

- Setup flexbox 'playground'

<https://jsfiddle.net/NiekVandael/g95eh0mv/12/>



```
HTML ▼ CSS ▼  
1 <div id="container">  
2   <ul id="flex-container">  
3     <li>1</li>  
4     <li>2</li>  
5     <li>3</li>  
6     <li>4</li>  
7     <li>5</li>  
8     <li>6</li>  
9   </ul>  
10 </div>  
  
1 ul #flex-container{  
2   list-style-type: none;  
3   margin: 0px;  
4   padding: 0px;  
5 }  
6  
7 ul li {  
8   height: 70px;  
9   padding-top: 10px;  
10  color: white;  
11  font-weight: bold;  
12  font-size: 1em;  
13  text-align: center;  
14 }  
15  
16 ul li:first-child {  
17   background: #c2d775;  
18 }  
19  
20 ul li:nth-child(2){  
21   background: #007278;  
22 }
```

# Display: flex

Probleem: Hoe krijgen we al deze items nu naast elkaar?

Oplossing: Container: → **display: flex**



```
CSS ▼  
1 ul#flex-container{  
2   list-style-type: none;  
3   margin: 0px;  
4   padding: 0px;  
5  
6   display: flex;  
7  
8 }  
9
```

# Flex-grow

Probleem: Hoe kunnen we het scherm 100% responsive vullen? (width)

Oplossing: children → **flex-grow: 1**

Viewport aanpassen (browser resize) = scaled mee!

```
8 ul li {  
9  
10 flex-grow: 1;  
11  
12 height: 70px;  
13 padding-top: 10px;  
14 color: white;  
15 font-weight: bold;  
16 font-size: 1em;  
17 text-align: center;  
18 }
```

# Flex-grow

Probleem: hoe kunnen we 1 child groter maken dan een ander?

Oplossing: child → **flex-grow: n**

```
20 ul li:first-child { background: #c2d775; }
21 ul li:nth-child(2){ background: #007278; }
22 ul li:nth-child(3){ background: #c7b600; flex-grow: 2; }
23 ul li:nth-child(4){ background: #a90077; }
24 ul li:nth-child(5){ background: #00a378; flex-grow: 5; }
25 ul li:last-child { background: #00ea72; }
```

# Order

Probleem: Hoe kan ik de volgorde van de items wijzigen?

Oplossing: children → order-property

```
20 ul li:first-child { background: #c2d775; order: 6; }
21 ul li:nth-child(2){ background: #007278; order: 5; }
22 ul li:nth-child(3){ background: #c7b600; order: 4; }
23 ul li:nth-child(4){ background: #a90077; order: 3; }
24 ul li:nth-child(5){ background: #00a378; order: 2; }
25 ul li:last-child  { background: #00ea72; order: 1; }
```

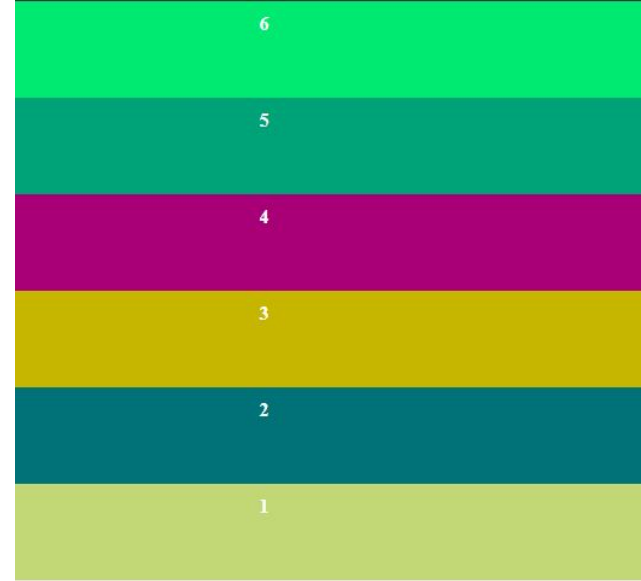
# Flex-direction

Probleem: de layout horizontaal/verticaal uitlijnen

Oplossing: container →

**flex-direction: row | column | row-reverse | column-reverse**

- row: elementen op 1 lijn weergeven (naast elkaar)
- column: 1 element per lijn weergeven (onder elkaar)
- row/column – reverse: elementen omgekeerd weergeven

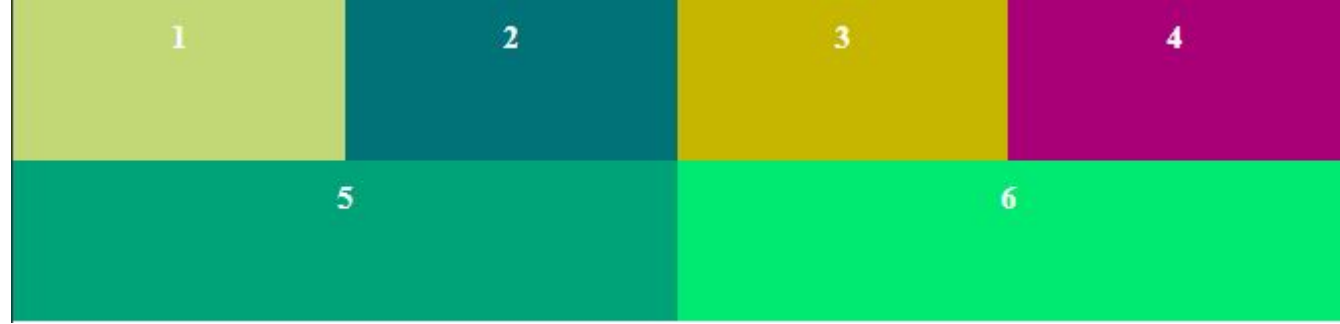


```
1 ul#flex-container{  
2   list-style-type: none;  
3   margin: 0px;  
4   padding: 0px;  
5   display: flex;  
6  
7   flex-direction: column-reverse;  
8 }
```

<https://jsfiddle.net/NiekVandael/g95eh0mv/26/>



# Flex-wrap



Probleem: items onder elkaar weergeven indien een minimale grootte niet meer getoond kan worden (scherm te klein)

Oplossing: Container → **flex-wrap: wrap**

Children → width: ...;

```
1 ul#flex-container{
2   list-style-type: none;
3   margin: 0px;
4   padding: 0px;
5   display: flex;
6   flex-direction: row;
7
8   flex-wrap: wrap;
9 }
```

```
11 ul li {
12
13   width: 150px;
14
15   flex-grow: 1;
16   height: 70px;
17   padding-top: 10px;
18   color: white;
19   font-weight: bold;
20   font-size: 1em;
21   text-align: center;
22 }
```



# Justify-content: flex-end/center

Probleem: items uitlijnen aan de rechts kant / centereren

Oplossing: container → justify-content: flex-end

- flex-end: items rechts uitlijnen
- center: items in het midden uitlijnen

```
1 ul#flex-container{  
2   list-style-type: none;  
3   margin: 0px;  
4   padding: 0px;  
5   display: flex;  
6   flex-direction: row;  
7   flex-wrap: wrap;  
8  
9   justify-content: flex-end;  
10 }
```

1

2

3

4

5

6

# Justify-content: space-between/space-around

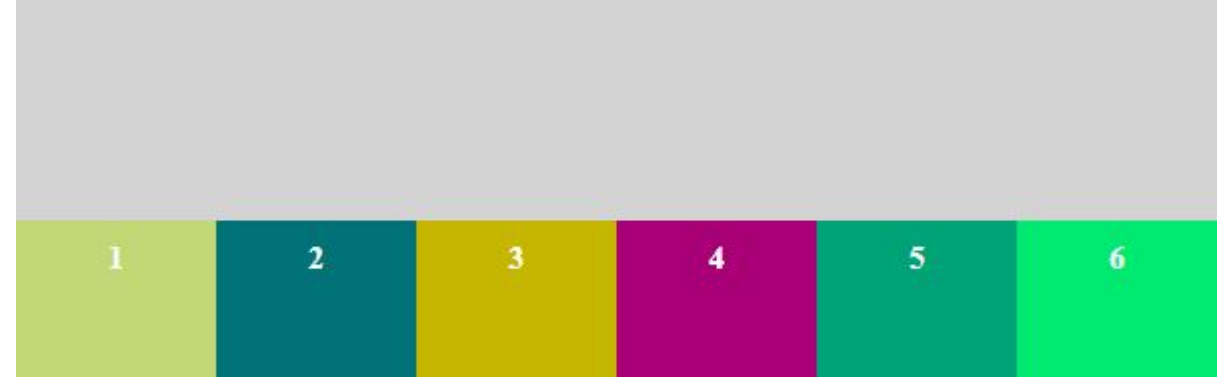
Probleem: items uitlijnen over de volledige width

Oplossing: container → **justify-content: space-between;**

- space-between: volledige pagina vullen, overige ruimte opvullen
- space-around: volledige pagina vullen, overige ruimte opvullen (ook links en rechts van het eerste & laatste item)

```
1 ul#flex-container{
2   list-style-type: none;
3   margin: 0px;
4   padding: 0px;
5   display: flex;
6   flex-direction: row;
7   flex-wrap: wrap;
8
9   justify-content: space-between;
10 }
```

# align-items: flex-end /center



Probleem: items verticaal uitlijnen (in het midden/onderaan)

Oplossing: container → align-items: flex-end;

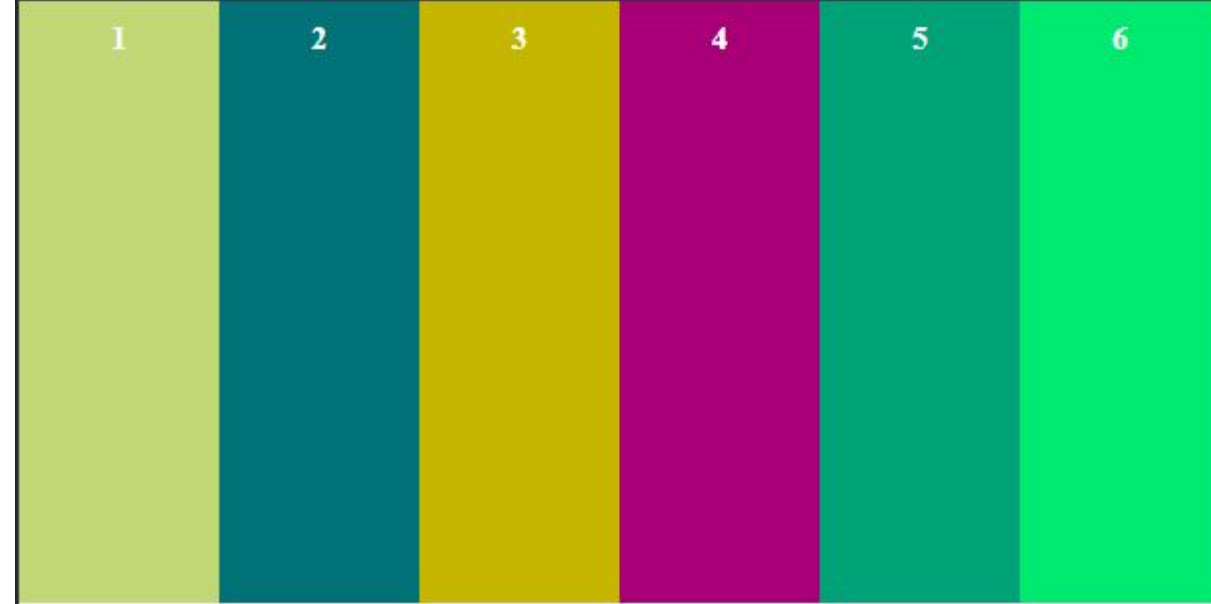
- flex-end: onderaan de container
- center: in het midden van de container

```
1 ul#flex-container{
2   list-style-type: none;
3   margin: 0px;
4   padding: 0px;
5   display: flex;
6   flex-direction: row;
7   flex-wrap: wrap;
8
9   height: 300px;
10  background-color: lightgray;
11
12  align-items: flex-end;
13 }
```

# align-items: stretch

Probleem: items de hoogte  
laten aannemen v/d parent

Oplossing: **align-items: stretch;**



```
1 ul#flex-container{  
2   list-style-type: none;  
3   margin: 0px;  
4   padding: 0px;  
5   display: flex;  
6   flex-direction: row;  
7   flex-wrap: wrap;  
8   height: 300px;  
9   align-items: stretch;  
10 }
```

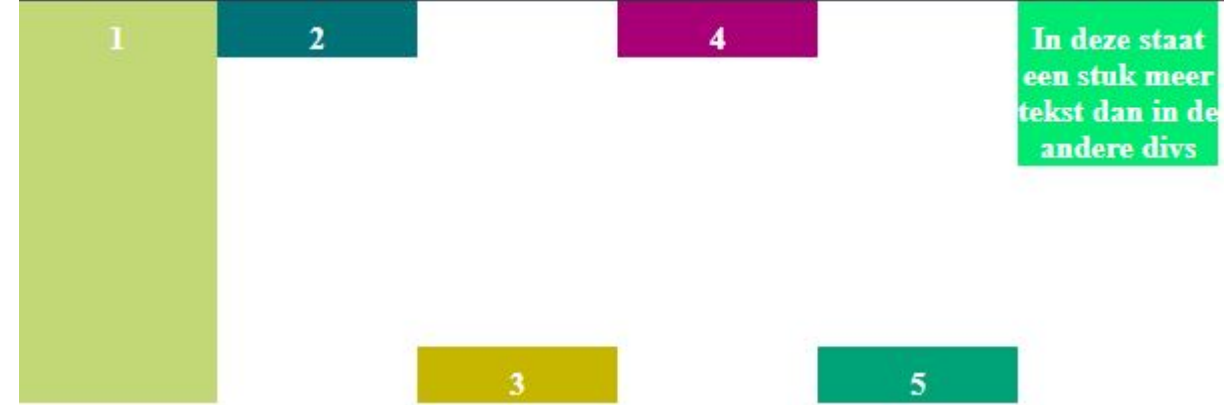
# align-items: baseline

Probleem: de hoogte van het child = de hoogte v/d content

Oplossing: container → **align-items: baseline;**

```
1  ul#flex-container{
2    list-style-type: none;
3    margin: 0px;
4    padding: 0px;
5    display: flex;
6    flex-direction: row;
7    flex-wrap: wrap;
8    height: 300px;
9
10   align-items: baseline;
11 }
```

# align-self: uitlijning per child



Probleem: uitlijning specificeren per child-element

Oplossing: child → **align-self: flex-end;**

- flex-end
- baseline
- stretch
- Center

> align-self niet aanwezig?

→ neem waarde van align-items op container

> align-items ook niet aanwezig?: stretch

```
1 ul#flex-container{
2   list-style-type: none;
3   margin: 0px;
4   padding: 0px;
5   display: flex;
6   flex-direction: row;
7   flex-wrap: wrap;
8   height: 200px;
9
10  align-items: stretch; /* default */
11 }
12
13 ul li {
14   width: 100px;
15   padding-top: 10px;
16   color: white;
17   font-weight: bold;
18   font-size: 1em;
19   text-align: center;
20 }
21
22 ul li:first-child { background: #c2d775; }
23 ul li:nth-child(2){ background: #007278; align-self: flex-start;}
24 ul li:nth-child(3){ background: #c7b600; align-self: flex-end;}
25 ul li:nth-child(4){ background: #a90077; align-self: baseline;}
26 ul li:nth-child(5){ background: #00a378; align-self: flex-end;}
27 ul li:last-child { background: #00ea72; align-self: baseline;}
```

<https://jsfiddle.net/NiekVandael/g95eh0mv/49/>

# Oefening: responsive gallery

Maak volgende pagina na met flexbox en mediaquery

