

Web development 1 – Verslag Labo 8

Bachelor in de **toegepaste informatica**

campus **Kortrijk**

Thomas De Groote

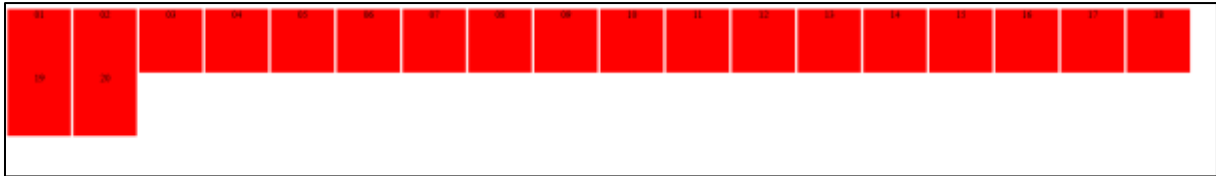
docent: **Pim Debaere**

academiejaar **2025–2026**

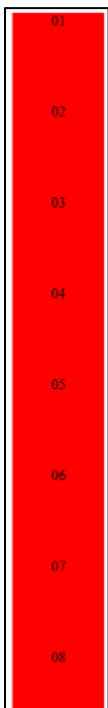
Opdracht 1

Dit is op de site te zien als in de class de display staat op:

```
display:inline-block;
```



Zoals te zien is staan alle rode objecten allemaal op 1 lijn, vanaf er geen plaats meer is gaan de objecten naar de volgende lijn.



Links hiernaast zien we het resultaat als:

```
display:block;
```

Alle objecten staan nu onder elkaar en niet meer op 1 lijn.

Als: `display:inline;`

Dan zijn de objecten geen 'block' meer maar staan ze wel op dezelfde lijn. De width en height uit de CSS hebben geen invloed meer.

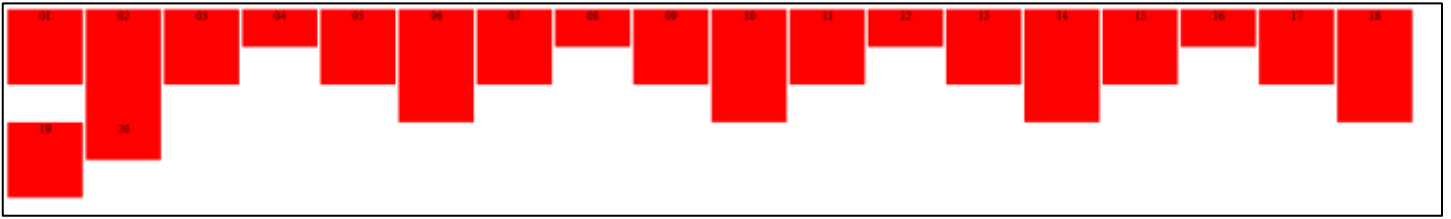
Resultaat:

01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20

Opdracht 2

De witte ruimte tussen de rode objecten zoals op de eerste afbeelding te zien komt omdat er in HTML standaard een spatie wordt gezet als 2 objecten op een andere lijn staan. Dit is op verschillende manieren weg te werken. Je kan bijvoorbeeld alles op 1 lijn zetten, wat niet heel praktisch is, of je kan bijvoorbeeld de spaties in commentaar zetten.

Opdracht 3



In voorbeeld2 zie je dat de objecten gecentreerd staan omdat in het <div>-element de tekst-align op 'center' staat, wat ook invloed heeft op de objecten omdat het inline-content is.

`vertical-align: center` bestaat niet, dus heeft sowieso geen invloed. Vertical-align werkt ook niet op block-level elementen zoals een <div>. Het werkt bv. wel op inline, inline-block.

`vertical-align: middle` centreert de tekst niet, het heeft geen invloed op de inhoud, het bepaalt enkel de uitlijning van het element tegenover andere inline-elementen.

Om tekst op een simpele manier te centreren kan je de line-height gebruiken.

```
.area {  
  height:100px;  
  line-height:100px; /* tekst staat nu precies in het midden */  
}
```

Om meerdere regels te centreren kan je de display aanpassen naar table-cell, hier kan wel verticaal gecentreerd worden.

```
.area {  
  display: table-cell;  
  vertical-align: middle;  
}
```

.tall krijgt voorrang omdat het later in CSS file voorkomt. Dit is makkelijk te testen door ze eens om te wisselen in de CSS-file. Het resultaat is dan dat er geen .tall elementen meer zijn, want die staat nu voor de klasse area.



Opdracht 4

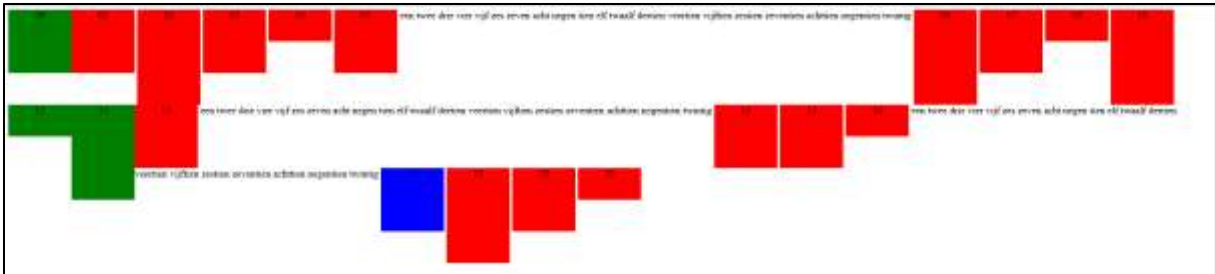
Standaard zonder aanpassingen:



Hier zijn enkele elementen floaty, dus met `float: left;` de elementen zweven nu links als het ware met alle andere elementen die errond zweven.

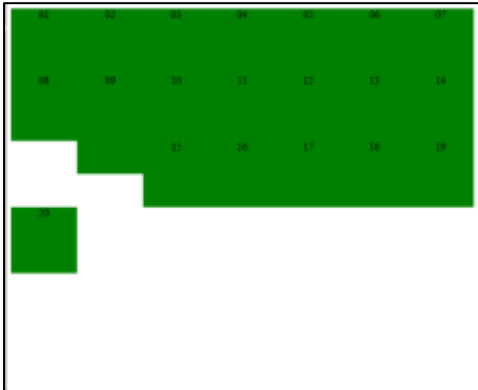
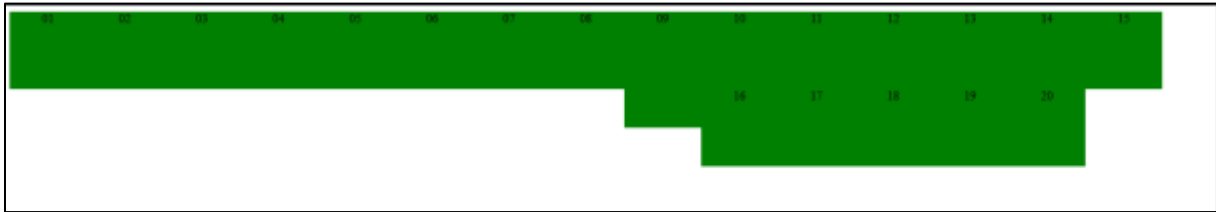


Hier heeft element 17 `break: bold;`



Dit zet het element onder alle andere floats, ongeacht of ze links of rechts zweven.

Opdracht 5



Als de pagina geresized wordt, dan kunnen verschillende elementen verplaatsen van lijn tot lijn.

Met float laat je elementen naast elkaar zweven zolang er ruimte is. In voorbeeld4.html zie je dat er naast element 09 meerdere rijen blokken komen wanneer het venster breed genoeg is.

Wanneer je element 13 de class break geeft (`clear: both;`), begint dit element altijd op een nieuwe regel, omdat clear afdwingt dat het onder alle gefloatte elementen moet staan.

Opdracht 6 – Lorem Ipsum

```
div {
  display: inline-block; /* Zet elk div naast elkaar zoals tekst */
  border: 1px solid black; /* Randje rond elk div */
  width: 50%; /* Elk div neemt de helft van de breedte */
  box-sizing: border-box; /* Rand telt mee in die 50% */
  margin-right: -2px; /* Verwijdert kleine witruimte tussen inline-blocks */
}
```

Om ervoor te zorgen dat 2 elementen exact 50% innemen gebruik je box-sizing om ervoor te zorgen dat de rand meetelt in de 50% van de width.

Opdracht 7 – Lorem Ipsum Extra

Om 10px witruimte rond te elementen toe te voegen doe je dit als volgt:

```
div {
  display: inline-block;
  width: calc(50% - 29px);
  margin: 0 10px;
  border: solid black 1px;
  box-sizing: border-box;
}
```

De margin-right staat op 10px en de width is met 29px verminderd (plaats voor de witruimte)

Opdracht 8 – Cocktailbar

CSS:

```
body {
  font-family: Verdana, Geneva, Tahoma, sans-serif;
  font-size: 1.2em;
  margin: 0 2.5% 2.5% 2.5%;
  display: flex;
}

header p img {
  display: block;
  margin: auto;
}

h1 {
  color: #007e7e;
  font-size: 150%;
}

h2 {
  font-size: 130%;
  color: #007e7e;
}

.elixirs {
  float: right;
  width: 30%;
  text-align: center;
  margin-right: 5px;
}

.elixirs a:hover {
  background: #7a593f;
}

#lemonbreeze {
  color: #efc93a;
}

#chairchiller {
  color: #7a593f;
}

#blackbrainbrew {
  color: #555563;
}

#content {
  width: 70%;
}

#guarantee {
  border: dotted gray 2px;
  margin: 30px;
  padding: 20px 20px 20px 80px;
  color: #444444;
  background-color: #a7cece;
  background-image: url("../images/background.gif");
  background-repeat: no-repeat;
}

footer {
  font-size: 0.5em;
}
```

