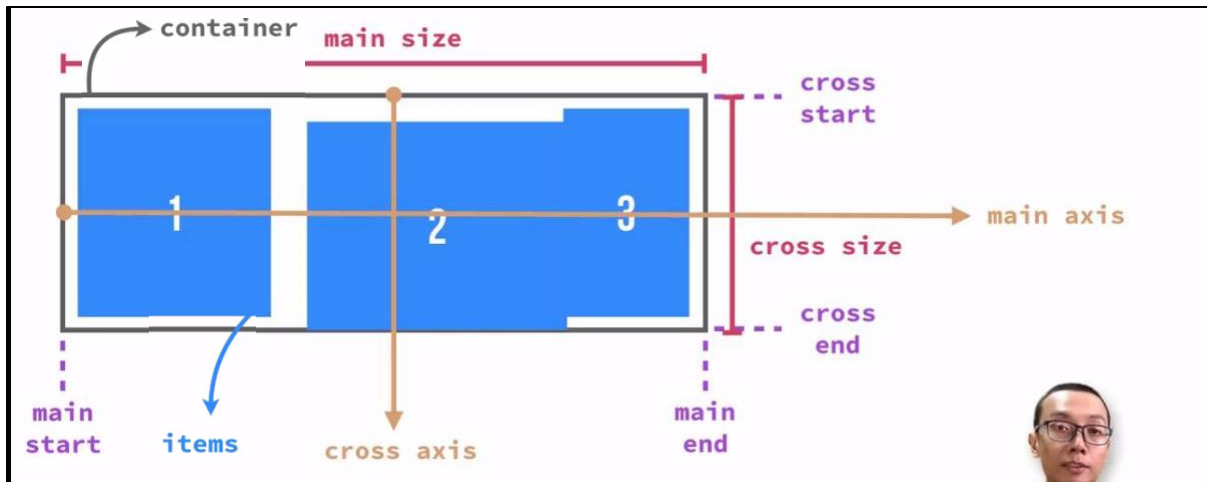


# Daftar Isi

<b>FLEXBOX .....</b>	<b>3</b>
<b>FLEXBOX CONTAINER PROPERTY .....</b>	<b>3</b>
▪ Basic .....	3
▪ display .....	3
▪ flex-direction .....	3
▪ flex-wrap .....	3
▪ justify-content .....	4
▪ align-items .....	4
▪ align-content .....	4
<b>FLEXBOX ITEMS PROPERTY .....</b>	<b>5</b>
▪ order .....	5
▪ flex-grow / flex .....	5
▪ align-self .....	5
<b>GRID .....</b>	<b>6</b>
<b>GRID CONTAINER PROPERTY .....</b>	<b>6</b>
▪ Basic .....	6
▪ Terminology .....	6
▪ grid-template-columns .....	6
▪ grid template-rows .....	6
▪ grid-auto-columns .....	6
▪ grid-auto-rows .....	6
▪ grid-auto-flow .....	6
▪ grid-template-areas .....	6
▪ grid-template .....	7
▪ column-gap .....	7
▪ row-gap .....	7
▪ gap .....	7
▪ grid .....	7
▪ justify-items .....	7
▪ align-items .....	7
▪ place-items .....	7
▪ justify-content .....	7

▪ align-content .....	7
▪ place-content.....	7
<b>GRID ITEMS PROPERTY .....</b>	<b>8</b>
▪ grid-column-start .....	8
▪ grid-column-end .....	8
▪ grid-row-start .....	8
▪ grid-row-end .....	8
▪ grid-column .....	8
▪ grid-row .....	8
▪ grid-area .....	8
▪ justify-self .....	8
▪ align-self .....	8
▪ place-self .....	8

# FLEXBOX



**\*Note :** property `display` sampai `align-content` adalah properti pada container.

## ■ Basic

- Struktur CSS : `selector { property: value; }`
- `box-sizing: border-box;` : Agar wadah container tidak berubah ukurannya.

## ■ `display`

value :

- `flex`
- `inline-flex`

## ■ `flex-direction`



Untuk mengatur arah dari items2 didalam container.

- `row` : kiri ke kanan
- `row-reverse` : kanan ke kiri
- `column` : atas ke bawah
- `column-reverse` : bawah ke atas

## ■ `flex-wrap`







Untuk mengatur jika items tidak muat di container (seperti alinea baru).

Jika tidak pakai wrap maka items akan diperkecil sizenya untuk memuatkan semua items.

- `nowrap` :  tidak memberi wrap
- `wrap` :  memberi wrap
- `wrap-reverse` : memberi wrap dan membalikan dari items terakhir dahulu






## ▪ justify-content

Untuk mengatur rata dari items2 di dalam container secara horizontal.

- flex-start : 
- flex-end : 
- center : 
- space-between : 
- space-around : 
- space-evenly : 

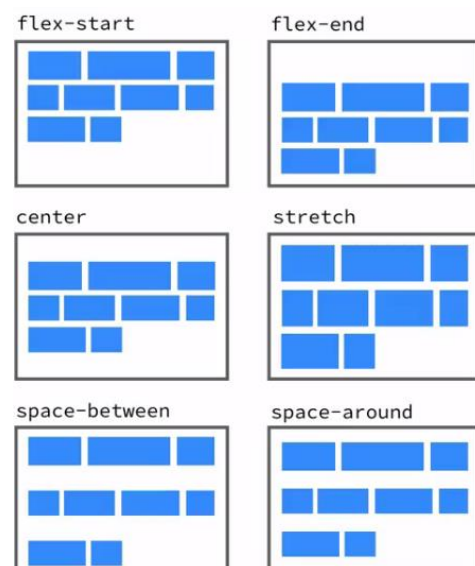
## ▪ align-items

Untuk mengatur rata dari items2 di dalam container secara vertikal.

- flex-start : 
- flex-end : 
- center : 
- stretch :  (jangan gunakan height pada items)
- baseline : 

## ▪ align-content

Sama seperti justify content, tetapi secara vertikal



```
.container {
  align-content: flex-start | flex-end | center |
                space-between | space-around;
}
```

Mengatur jarak antar items terhadap cross axis  
(hanya berfungsi ketika items lebih dari 1 baris)



**\*Note :** property **display** sampai **align-content** adalah properti pada container.

**\*Note :** property `order`, `flex-grow`, dan `align-self` adalah properti pada items.


- **order**

Untuk mengatur urutan items2nya.

```
<div class="container">
  <div class="satu">1</div>
  <div class="dua">2</div>
  <div class="tiga">3</div>
  <div class="empat">4</div>
</div>
```




```
.satu { order: 2; }
.dua { order: 3; }
.tiga { order: 4; }
.empat { order: 1; }
```




- **flex-grow / flex**

Untuk mengatur ukuran width items. dan height items mengikuti height container.

```
.satu { flex-grow: 1; }
.dua { flex-grow: 1; }
.tiga { flex-grow: 1; }
.empat { flex-grow: 1; }
```




```
.satu { flex-grow: 1; }
.dua { flex-grow: 3; }
.tiga { flex-grow: 1; }
.empat { flex-grow: 1; }
```



- **align-self**

Untuk mengatur hanya satu item untuk diposisikan ke posisi yg diinginkan.

flex-start



stretch

flex-end

```
.item {
  align-self: auto | flex-start | flex-end |
              center | baseline | stretch;
}
```

**\*Note :** property `order`, `flex-grow`, dan `align-self` adalah properti pada items.

# GRID

## ■ Basic

- Struktur CSS : `selector { property: value; }`
- `box-sizing: border-box;` : Agar wadah container tidak berubah ukurannya.

## ■ Terminology

- Grid container : Elemen pembungkus grid [`display: grid;`]
- Grid item : Elemen2 yg berada (1level) di dalam grid container.
- Grid line : Garis horizontal (kolom) atau vertikal (baris) yg memisahkan grid menjadi beberapa bagian dan ditandai dengan angka.
- Grid cell : Perpotongan / pertemuan antara baris dan kolom dalam grid.
- Grid area : Kumpulan lebih dari satu grid cell yg membentuk kotak.
- Grid track : Ukuran / jarak antara 2 grid line, bisa horizontal/vertikal.
- Grid gap : Spasi / jarak antar grid track/cell.

## ■ `grid-template-columns`

## ■ `grid-template-rows`

Mendefinisikan jumlah kolom dan baris

value :

- `px, %, auto, fr (fraction),` atau `minmax()`
- `repeat(9, 1fr)` : `repeat(jml/auto-fill/auto-fit, size);`
- `min-content` : width berukuran 1 kata yg terpanjang
- `max-content` : width berukuran seluruh kata
- `minmax()` : `minmax(minwidth, maxwidth)`

## ■ `grid-auto-columns`

## ■ `grid-auto-rows`

Mengatur ukuran grid track yg tidak dituliskan pada `grid-template` (implicit track)

- `px, %, auto,` atau `fr (fraction)`

## ■ `grid-auto-flow`

Untuk mengatur arah dari item/cell didalam container/grid track.

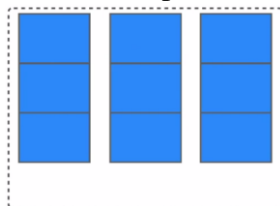
- `row` : atas ke bawah
- `row dense` : bawah ke atas
- `column` : kiri ke kanan
- `column dense` : kanan ke kiri

## ■ `grid-template-areas`

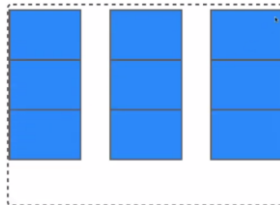
Membuat grid template menggunakan nama dari area yg ditulis pada property `grid-area` pada item.

- `grid-template-areas:` `"header header"`  
`"sidebar content"`  
`"footer footer";`

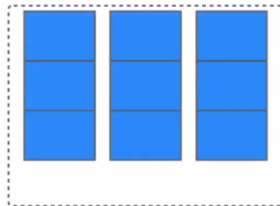
- **grid-template**  
Shorthand grid
- **column-gap**
- **row-gap**
- **gap**  
Membuat jarak cell grid secara horizontal(column), vertikal(row) atau keduanya(gap).
- **grid**  
Shorthand (jarang dipakai)
- **justify-items**
- **align-items**
- **place-items**  
Mensejajarkan grid-items pada sumbu horizon(justify) / vertikal(align) / keduanya(place).
  - start : kiri
  - end : kanan
  - center : tengah
  - stretch : full kiri sampai kanan
- **justify-content**
- **align-content**
- **place-content**  
Mengatur posisi seluruh grid container pada sumbu horizontal(justify) / vertikal(align) / keduanya(place).
  - start : kiri
  - end : kanan
  - center : tengah
  - stretch : full kiri sampai kanan



- space-around :



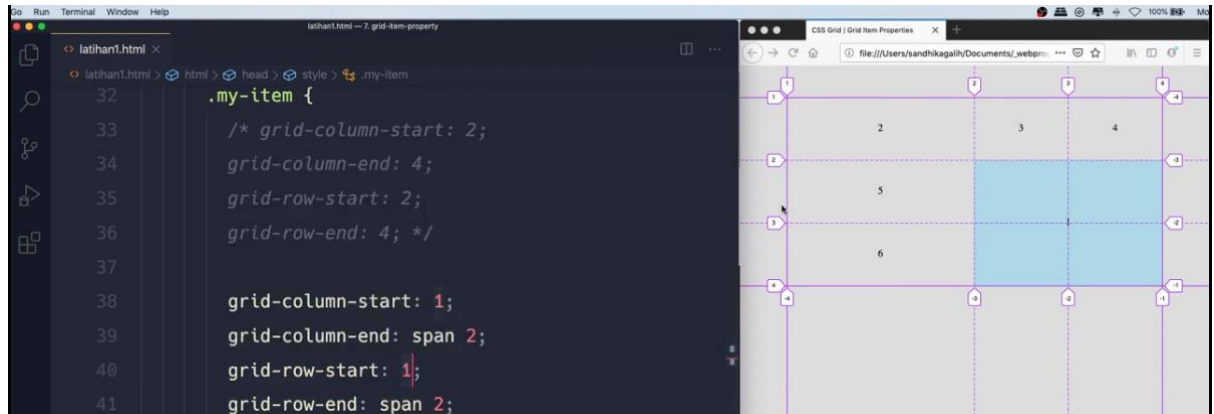
- space-between :



- space-evenly :

- **grid-column-start**
- **grid-column-end**
- **grid-row-start**
- **grid-row-end**

Menentukan tempat dari grid item dgn mengacu pada nomor/nama grid line yg spesifik.



- **grid-column**
- **grid-row**

Shorthand untuk start dan end pada grid column dan row.

- `grid-column: start end;`
- `grid-row: start end;`

- **grid-area**

Menentukan nama area sesuai properti `grid-template-areas`.

Dan

Bisa digunakan sebagai shorthand start end row column

- `grid area: row-start / column-start / row-end / column-end;`

```
grid-area: 2 / 1 / 4 / 3;
```

- **justify-self**

Mengatur posisi item pada sebuah cell terhadap sumbu horizontal.

- `start`, `end`, `center`, `stretch`.

- **align-self**

Mengatur posisi item pada sebuah cell terhadap sumbu vertikal.

- `start`, `end`, `center`, `stretch`.

- **place-self**

Shorthand `justify-self` dan `align-self`.

- `place-self: align-self justify-self;`