**FLEXBOX**

Sebuah cara yang menawarkan kita untuk menyusun dan mensejajarkan element di dalam sebuah container (parent) dengan baik.

Untuk menggunakan flexbox kita butuh sebuah ***flex-container*** yaitu sebuah *parent element*. Element didalam flex-container / childnya kita sebut dengan ***flex-items***.

1. **Membuat *flex-container***

Cari element parent yang menjadi container dari beberapa item. Beri element itu di css sebuah property **display: flex;**

1. **Memberikan jarak antar element**

Target *flex-container* (parent) di css dan beri property **gap: <angka><unit>**

**<angka>: 16**

**<unit>: px|%|rem|em, dan sebagainya**

**gap: 16px;**

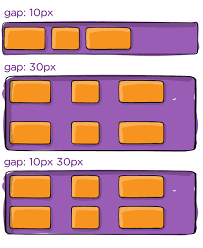
Bentuk penulisan ini adalah untuk memberi jarak secara horizontal dan vertikal dengan nilai sama.

Kita bisa memberi nilai terpisah:

**gap: <vertikal> <horizontal>**

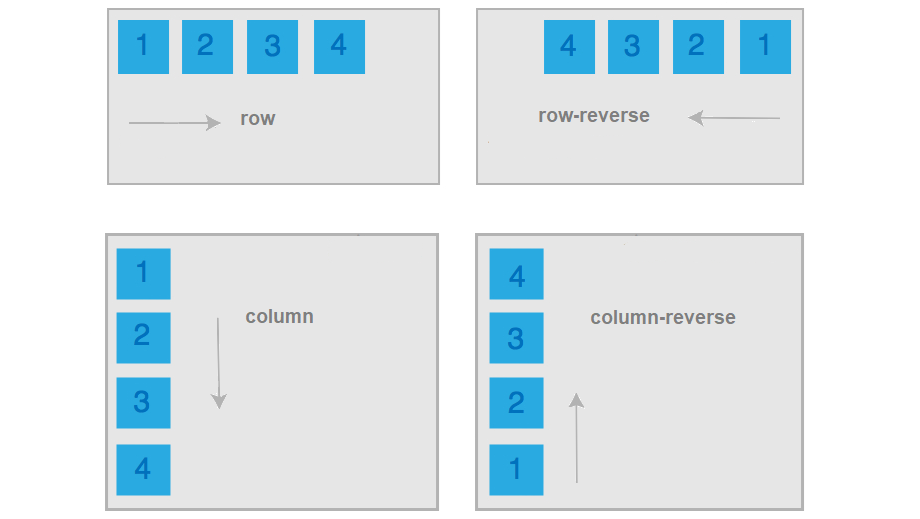
**gap: 10px 30px;**

Contoh:



1. **Memberikan *arah / direction* untuk *flex-container***

Target *flex-container* (parent) di css dan beri property **flex-direction: row | row-reverse | column | column-reverse;** (pilih salah satu diantara tanda **|**). Nilai default adalah **row.**

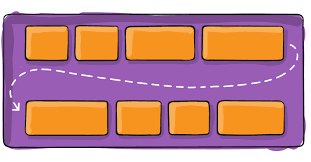


1. **Membuat *baris baru* untuk element yang tidak cukup dalam *flex-container***

Secara default, seluruh element di dalam *flex-container* akan dipaksa di satu baris saja, sehingga semua element akan menjadi gepeng (kehilangan width). Supaya element bisa turun ke baris baru JIKA TIDAK CUKUP dalam satu baris, maka:

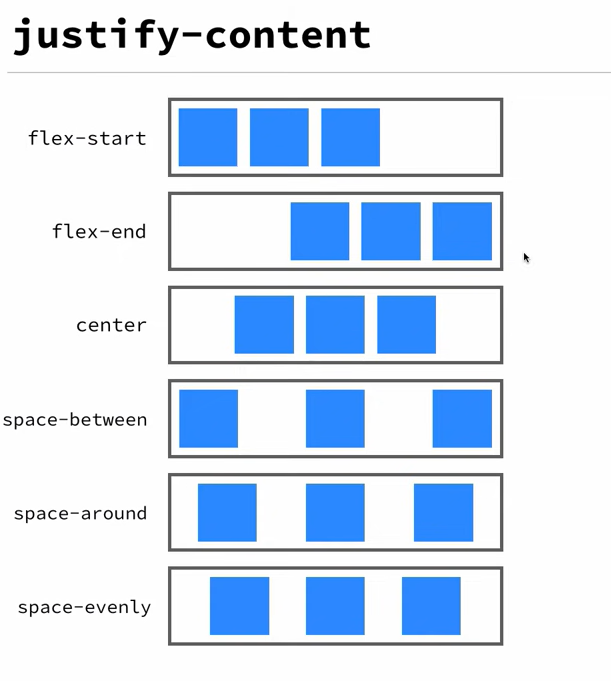
Target *flex-container* (parent), berikan property **flex-wrap: wrap;**

**Secara default nilainya flex-wrap: no-wrap;**

****

1. **Mengatur letak element (flex-item) di dalam *flex-container* secara HORIZONTAL*.***

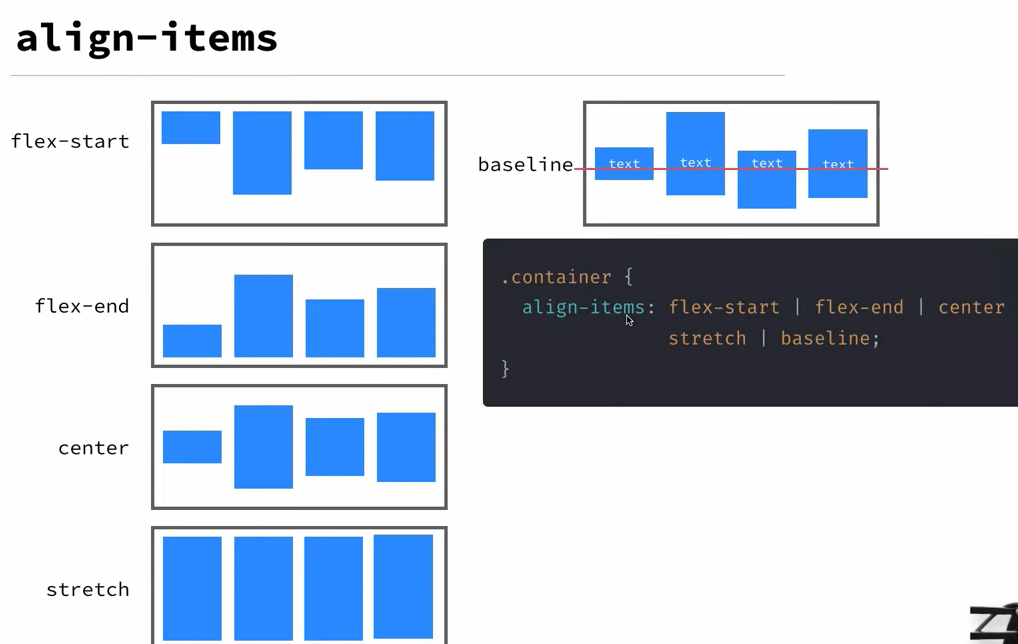
Target *flex-container* (parent) di css dan beri property **justify-content: flex-start | flex-end | center | space-between | space-evenly | space-around** (pilih salah satu diantara tanda **|**). Nilai default adalah **flex-start.**

****

****

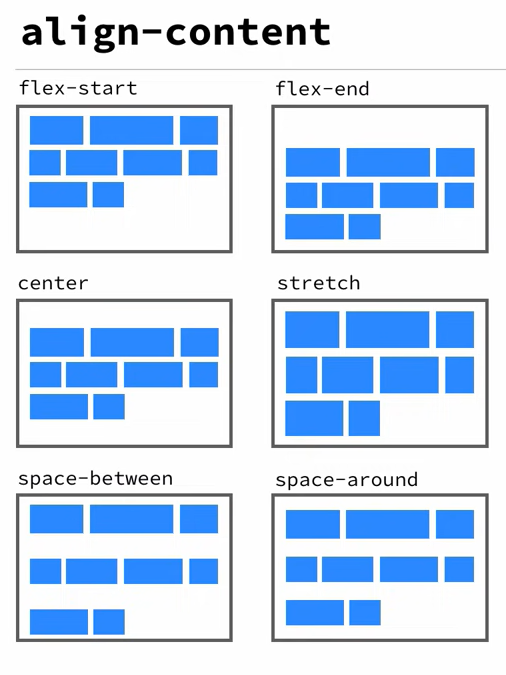
1. **Mengatur letak element (flex-item) di dalam *flex-container* secara VERTIKAL*.***

Target *flex-container* (parent) di css dan beri property **align-item: flex-start | flex-end | center | stretch | baseline** (pilih salah satu diantara tanda **|**). Nilai default adalah **stretch.**

****

1. **Mengatur letak element (flex-item) di dalam *flex-container* secara VERTIKAL jika *ITEMS LEBIH DARI 1 BARIS.***

Target *flex-container* (parent) di css dan beri property **align-content: flex-start | flex-end | center | stretch | space-between | space-around | space-evenly** (pilih salah satu diantara tanda **|**). Nilai default adalah **stretch.**

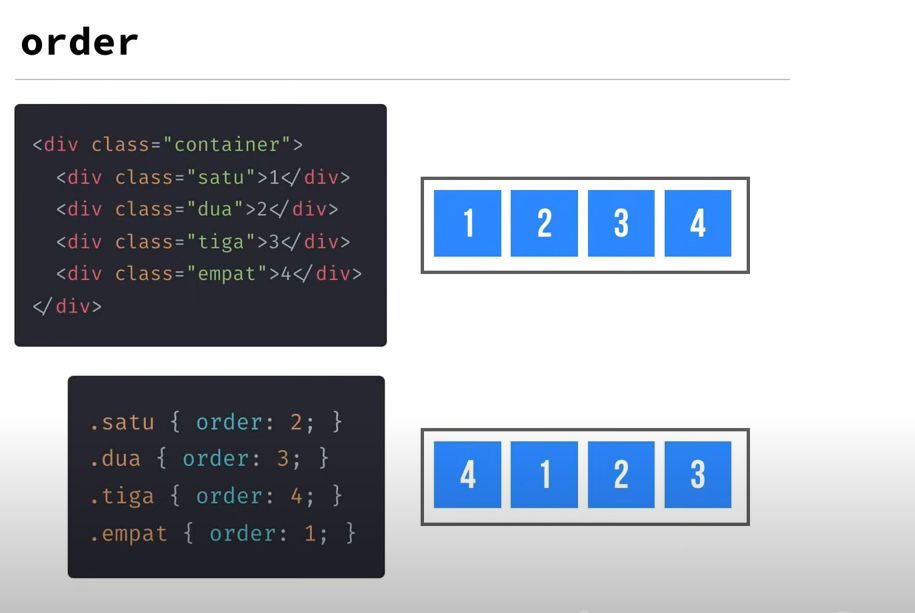
****

****

1. **Mengatur urutan / order dari *flex-items atau child* dari *flex-container.* (PROPERTY PADA CHILD)**

Target *flex-itemnya* lalu beri property **order: <nilai>;** contoh – *order: 2;*

Nilai defaultnya 0

****

1. **Mengatur ukuran / size dari *flex-items atau* *child* terhadap flex-container. (PROPERTY PADA CHILD).**

Target *flex-itemnya* lalu beri property **flex: <angka>;** contoh – ***flex: 2****; maka item ini akan membesar 2x dari element lainnya.*

Property ini adalah *short-hand* dari tiga property lainnya yaitu, ***flex-grow****,* ***flex-shrink****, dan* ***flex-basis.***

Yang perlu kita ketahui sekarang adalah **flex,** misalnya **flex: 1; artinya element akan membesar 1x terhadap element lainnya (ukurannya sama).**

****