

1. Tujuan:

- a. Mengetahui, Memahami dan menguasai penggunaan Flexbox layout

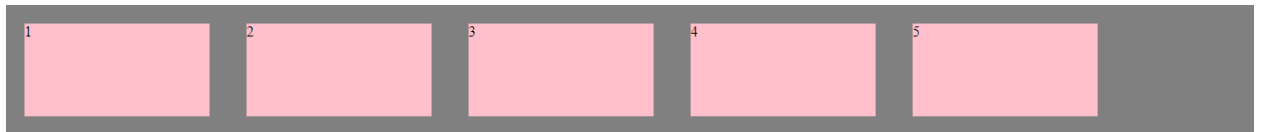
2. Materi:**2.1. Flexbox**

Flexbox merupakan mode layout terbaru yang menjadi fitur dari CSS3. Sebelumnya kita mengenal mode block, inline, table dan juga melibatkan floating untuk layouting. Float sebenarnya digunakan untuk memposisikan gambar/text.

Flexible box atau lebih dikenal dengan flexbox ini dapat mengatur ukuran child secara otomatis, dan mampu beradaptasi dengan ukuran kontainernya. Permasalahan-permasalahan yang sebelumnya diselesaikan dengan metode float, absolute dan transform, dapat diselesaikan secara mudah dengan menggunakan flexbox.

2.2 Flexbox Element

Untuk memulai menggunakan flexbox model, maka kita perlu mendefinisikan sebuah flex container. Di dalam flex container dapat diisi oleh beberapa flex item.



Flex container adalah elemen yang berwarna abu-abu, dan flex item adalah elemen yang berwarna pink. Untuk membuat flex container, maka kita tambahkan property:

`display: flex;`

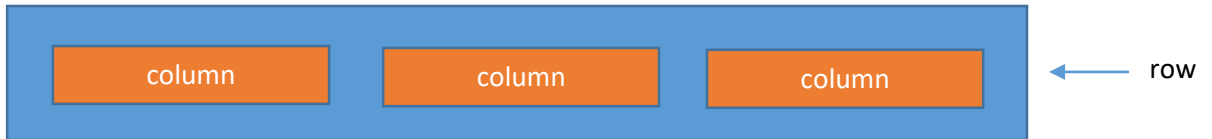
contoh:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     .container{
7       background-color: grey;
8       display: flex;
9     }
10  }
11  .container > .box{
12    width: 200px;
13    height: 100px;
14    margin: 20px;
15    background-color: pink
16  }
17 </style>
18 </head>
19 <body>
20   <div class="container">
21     <div class=" box ">1</div>
22     <div class=" box ">2</div>
23     <div class=" box ">3</div>
24     <div class=" box ">4</div>
25     <div class=" box ">5</div>
26   </div>
27 </body>
28 </html>
```

Ketika width dari flex item sudah didefinisikan dan ketika di implementasikan widthnya tidak cukup maka akan otomatis di overwrite.

2.3 Flexbox direction

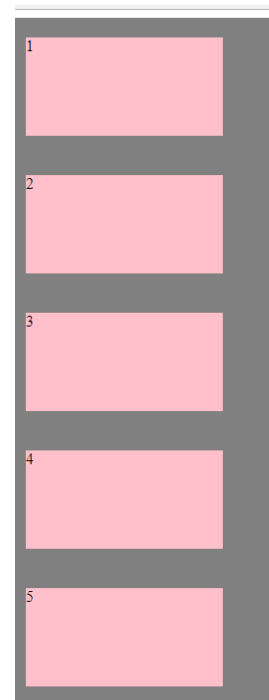
Secara default ketika menerapkan layout flex model pada container, maka elemen-elemen di dalamnya akan berlaku sebagai kolom, hal ini karna secara default



Untuk mengatur orientasi/arrah darai flex item dapat dilakukan dengan menambahkan property flex-direction, kemudian value nya dapat diisi dengan row atau column, pada flex containernya.

Contoh:

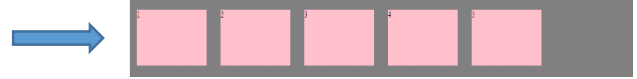
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     .container{
7       background-color: grey;
8
9       display: flex;
10      flex-direction: column;
11    }
12    .container > .box{
13      width: 200px;
14      height: 100px;
15      margin: 20px;
16      background-color: pink
17    }
18  </style>
19 </head>
20 <body>
21   <div class="container">
22     <div class=" box " >1</div>
23     <div class=" box " >2</div>
24     <div class=" box " >3</div>
25     <div class=" box " >4</div>
26     <div class=" box " >5</div>
27   </div>
28 </body>
29 </html>
```



```

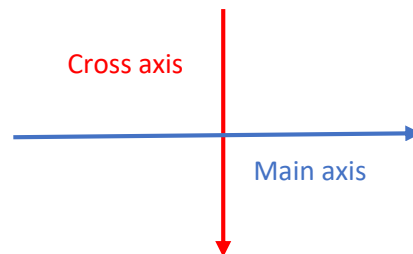
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     .container{
7       background-color: grey;
8       display: flex;
9       flex-direction: row;
10    }
11    .container > .box{
12      width: 200px;
13      height: 100px;
14      margin: 20px;
15      background-color: pink
16    }
17  </style>
18 </head>
19 <body>
20   <div class="container">
21     <div class=" box ">1</div>
22     <div class=" box ">2</div>
23     <div class=" box ">3</div>
24     <div class=" box ">4</div>
25     <div class=" box ">5</div>
26   </div>
27 </body>
28 </html>

```



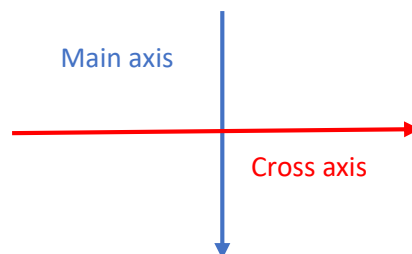
2.4 Flexbox Axis

Dalam flexbox dikenal 2 sumbu/axis, yakni main axis dan cross axis.



Ketika **flex-direction** diberi value **row**, maka **main axis** nya dari **kiri ke kanan** dan **cross axis** nya **dari atas ke bawah**.

Ketika **flex-direction** diberi value **column**, maka **main axis** nya dari **atas ke bawah** dan **cross axis** nya **kiri ke kanan**.



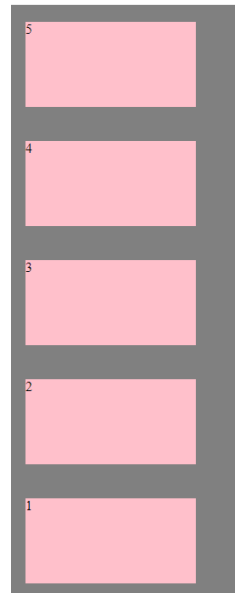
Untuk mengubah arah dari main axis dan cross axis maka dapat dilakukan dengan mengubah nilai dari **flex-direction: row-reverse** atau

flex-direction: column-reverse

```
.container{
  background-color: grey;
  display: flex;
  flex-direction: row-reverse;
}
```



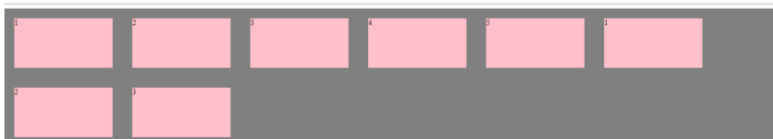
```
.container{
  background-color: grey;
  display: flex;
  flex-direction: column-reverse;
}
```



2.5 Flex Wrap

Property flex-wrap digunakan untuk mengatur tampilan sesuai dengan width yang telah ditentukan oleh flex item. Ketika lebih dari ukuran container nya, maka boxnya akan turun ke bawah. Flex-wrap secara default berisi no-wrap.

Contoh penggunaan:



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     .container{
7       background-color: grey;
8       display: flex;
9       flex-wrap: wrap;
10    }
```

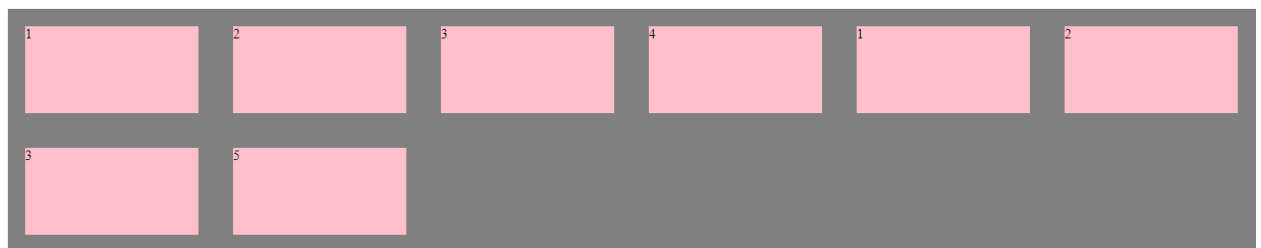
2.6 Pengaturan Flex Item

2.6.1 Mengurutkan Flex Item

Untuk mengurutkan tataletak dari flex item, bias dilakukan dengan mengatur property `order` dan value nya diisi angka boleh positive boleh negative.

Contoh:

```
<title></title>
<style type="text/css">
  .container{
    background-color: grey;
    display: flex;
    flex-wrap: wrap;
  }
  .container > .box{
    width: 200px;
    height: 100px;
    margin: 20px;
    background-color: pink
  }
  .box5{
    order: 1;
  }
</style>
```



2.6.1 Meratakan Flex Item

Untuk meratakan flex item berdasarkan **main axis** bias dilakukan dengan menggunakan property **justify-content** pada bagian flex container nya. Adapun value nya ada beberapa opsi: **flex-start** | **flex-end** | **center** | **space-between** | **space-around**

property	value	kegunaan
Justify-content	flex-start	Meratakan flex item ke sebelah kiri
	flex-end	Meratakan flex item ke sebelah kanan
	center	Meratakan flex item ke bagian tengah
	space-between	Meratakan ruang kosong antara flex item yang satu dengan yang lain
	space-around	Meratakan ruang kosong antara flex item yang satu dengan yang lain, dengan membagi kekosongan container secara merata ke flex item nya

Contoh:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     .container{
7       background-color: grey;
8       display:flex;
9       flex-wrap:wrap;
10      justify-content: flex-end;
11    }
12    .container > .box{
13      width: 200px;
14      height:100px;
15      margin:20px;
16      background-color: pink
17    }
18  </style>
19 </head>
20 <body>
21   <div class="container">
22     <div class=" box ">1</div>
23     <div class=" box ">2</div>
24     <div class=" box ">3</div>
25   </div>
26 </body>
27 </html>
```

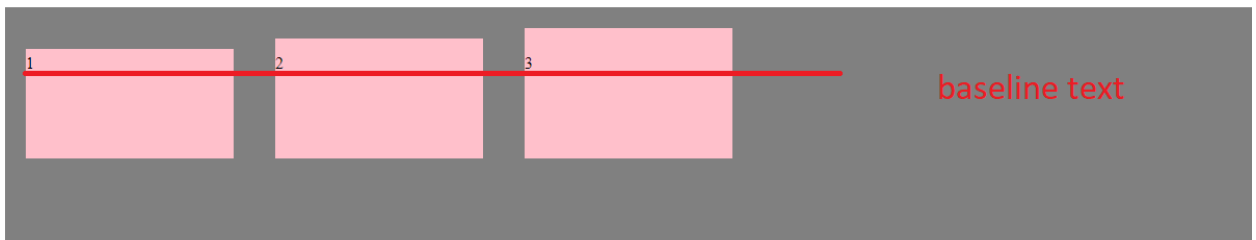


Untuk meratakan flex item berdasarkan **cross axis** bias dilakukan dengan menggunakan property **align-items** pada bagian flex container nya. Adapun value nya ada beberapa opsi: **stretch** | **flex-start** | **flex-end** | **center**|**baseline** |**space-around**

property	value	kegunaan
align-items	strech	Meratakan flex item sesuai tinggi container
	flex-start	Meratakan flex item ke sebelah atas
	flex-end	Meratakan flex item ke sebelah bawah
	center	Meratakan flex item ke bagian tengah
	baseline	Meratakan flex item berdasarkan posisi baseline text

Contoh:

```
<style type="text/css">
  .container{
    background-color: grey;
    display: flex;
    height: 100vh;
    align-items: baseline;
  }
  .container > .box{
    width: 200px;
    height: 100px;
    margin: 20px;
    background-color: pink
  }
  .box1{
    padding-top: 5px;
  }
  .box2{
    padding-top: 15px;
  }
  .box3{
    padding-top: 25px;
  }
</style>
</head>
<body>
  <div class="container">
    <div class=" box box1">1</div>
    <div class=" box box2">2</div>
    <div class=" box box3">3</div>
  </div>
</body>
```



Untuk meratakan flex item berdasarkan **cross axis**, jika flex item lebih dari 1 baris (**multiline**) bias dilakukan dengan menggunakan property **align-content** pada bagian flex container nya. Adapun value nya ada beberapa opsi: **stretch** | **flex-start** | **flex-end** | **center** | **baseline** | **space-between** | **space-around**

property	value	kegunaan
align-content	flex-start	Meratakan flex item ke sebelah atas
	flex-end	Meratakan flex item ke sebelah bawah
	center	Meratakan flex item ke bagian tengah
	space-between	Meratakan ruang kosong antara flex item yang satu dengan yang lain

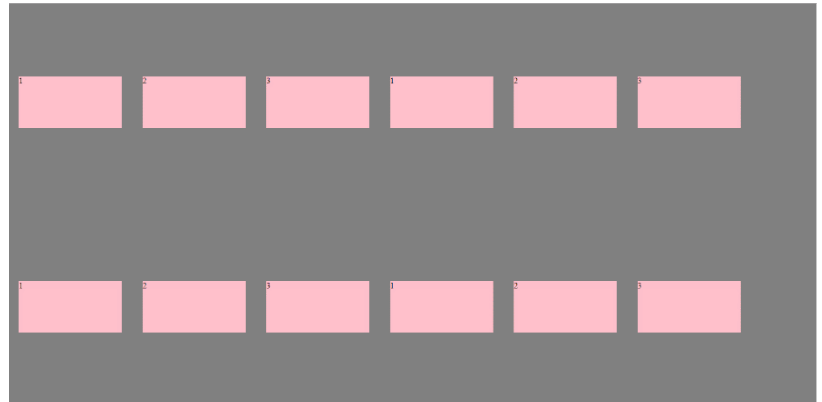
	space-around	Meratakan ruang kosong antara flex item yang satu dengan yang lain, dengan membagi kekosongan container secara merata ke flex item nya
--	--------------	--

Contoh:

```

4  <title></title>
5  <style type="text/css">
6    .container{
7      background-color: grey;
8      display: flex;
9      flex-wrap: wrap;
10     height: 100vh;
11     align-content: space-around; ;
12   }
13   .container > .box{
14     width: 200px;
15     height: 100px;
16     margin: 20px;
17     background-color: pink
18   }
19 </style>
20 </head>
21 <body>
22   <div class="container">
23     <div class=" box box1">1</div>
24     <div class=" box box2">2</div>
25     <div class=" box box3">3</div>
26     <div class=" box box1">1</div>
27     <div class=" box box2">2</div>
28     <div class=" box box3">3</div>
29     <div class=" box box1">1</div>
30     <div class=" box box2">2</div>
31     <div class=" box box3">3</div>
32     <div class=" box box1">1</div>
33     <div class=" box box2">2</div>
34     <div class=" box box3">3</div>
35   </div>
36 </body>

```



RANGKUMAN:

PROPERTI UNTUK FLEX CONTAINER

Property	Value
Display	Flex inline-flex
Flex-direction	Row row-reverse column column-reverse
Flex-wrap	Nowrap wrap stretch
Justify-content	Flex-start flex-end center space-between space-around
Align-items	Stretch flex-start flex-end center baseline
Align-content	Stretch flex-start flex-end center space-between space-around

PROPERTI UNTUK FLEX ITEM

Property	value
order	number