

Flex Box

THE MARKUP

```
<div id="container"> <div class="box box1">1</div> <div class="box box2">2</div> <div class="box box3">3</div> <div class="box box4">4</div> <div class="box box5">5</div></div>
```

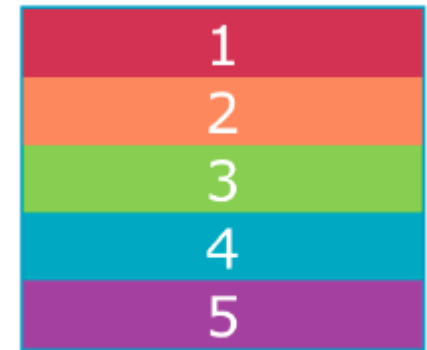
THE STYLES

```
#container {      display: flex;}
```

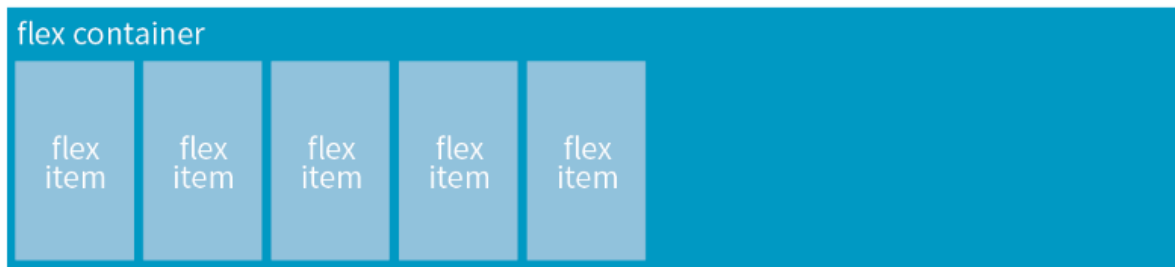
```
display: flex;
```



flexbox layout mode



block layout mode



در اینجا ۵ تگ div داریم که با اعمال استایل display: flex روی div نگهدارنده به حالت افقی نمایش داده می شوند

flex-direction

flex-direction

Values: row | column | row-reverse | column-reverse

Default: row

Applies to: flex containers

Inherits: no

دستوری برای تعیین جهت
آیتمهای flex

flex-direction: **row**; (default)



flex-direction: **row-reverse**;



flex-direction: **column**;



flex-direction: **column-reverse**;



flex-wrap

flex-wrap

Values: nowrap | wrap | wrap-reverse

Default: nowrap

Applies to: flex containers

Inherits: no

اگر مجموع اندازه آیتمهای فلکس بیشتر از اندازه عنصر نگهدارنده شود، به وسیله این ویژگی می توان کنترل کرد که آیتمهای فلکس به خط بعد بروند یا در همان خط بمانند و از کادر بیرون بزنند

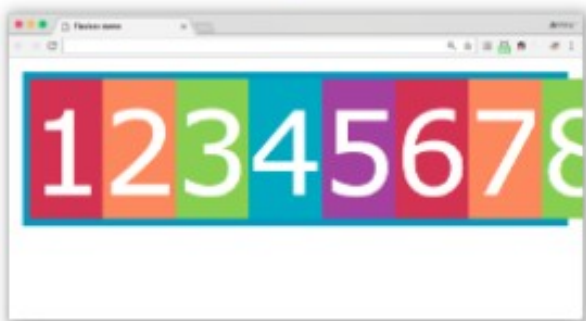
flex-wrap: wrap;



flex-wrap: wrap-reverse;



flex-wrap: nowrap; (default)



flex-flow

flex-flow

Values: *flex-direction flex-wrap*

Default: row nowrap

Applies to: flex containers

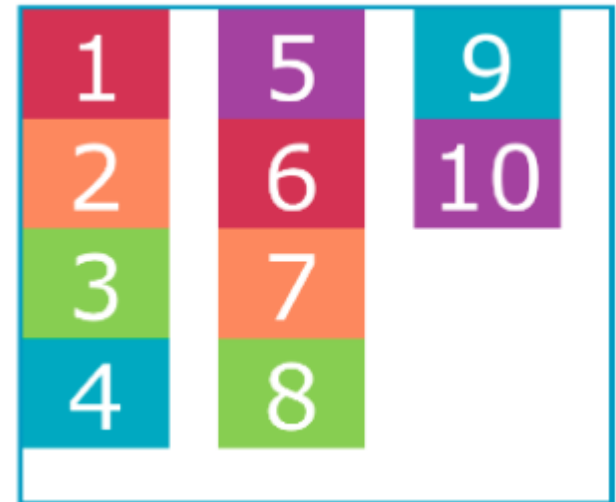
Inherits: no

با این دستور می‌توان مقادیر flex-direction , flex-wrap را معین نمود

```
#container { display: flex; height: 350px; flex-flow: column wrap;}
```

دستور بالا معادل است با

```
#container { display: flex; height: 350px; flex-direction: column; flex-wrap: wrap;}
```



justify-content

justify-content

Values: flex-start | flex-end | center | space-between | space-around

Default: flex-start

Applies to: flex containers

Inherits: no

`justify-content: flex-start;` (default)



`justify-content: flex-end;`



`justify-content: space-between;`



تعیین نحوه چینش در محور افقی (زمانی که جهت افقی باشد)
نکته : زمانی که جهت عمودی باشد چینش عمودی است

`justify-content: center;`



`justify-content: space-around;`



align-items

align-items

Values: flex-start | flex-end | center | baseline | stretch

Default: stretch

Applies to: flex containers

Inherits: no

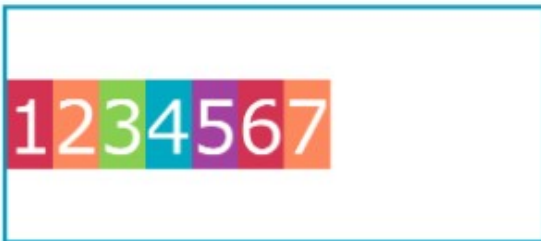
align-items: flex-start;



align-items: flex-end;



align-items: center;



align-items: stretch; (default)



align-items: baseline;



تعیین نحوه چینش در
محور عمودی (زمانی که
جهت افقی باشد)
نکته : زمانی که جهت
عمودی باشد چینش
افقی است

بر اساس موقعیت محتوای
اولین عنصر بقیه چینش
می‌شوند فارغ از اندازه شان

align-self

align-self

Values: flex-start | flex-end | center | baseline | stretch

Default: stretch

Applies to: flex items

Inherits: no

تعیین نحوه چینش در
محور عمودی (زمانی که
جهت افقی باشد)
برای یک عنصر

```
.box4 { align-self: flex-end;}
```

align-self: flex-end;



align-content

align-content

Values: flex-start | flex-end | center | space-around | space-between | stretch

Default: stretch

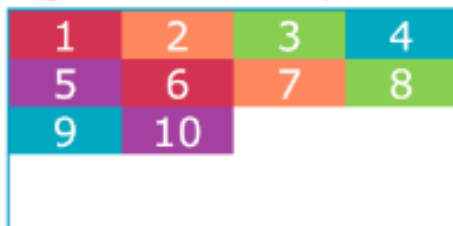
Applies to: multi-line flex containers

Inherits: no

تعیین نحوه چینش در
محور عمودی (زمانی که عناصر در
نگهدارنده بیشتر از یک خط باشند)

```
#container { display: flex; flex-direction: row; flex-wrap: wrap; height: 350px; align-items: flex-start;} box { width: 25%;}
```

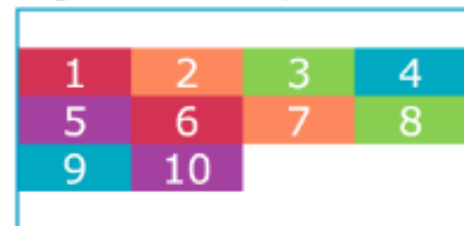
`align-content: flex-start;`



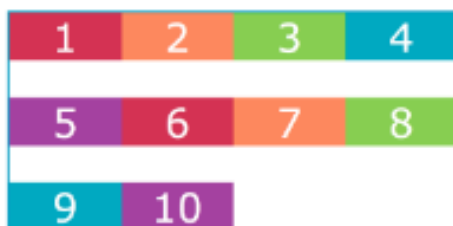
`align-content: flex-end;`



`align-content: center;`



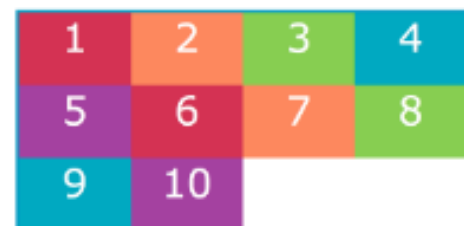
`align-content: space-between;`



`align-content: space-around;`



`align-content: stretch;` (default)



Flex, flex-grow

flex

Values: none | 'flex-grow flex-shrink flex-basis'

Default: 0 1 auto

Applies to: flex items

Inherits: no

روی آیتمهای فلکس اعمال می شود
اجازه و مقدار رشد -کاهش و مقدار
اولیه آیتم را مشخص می کند

flex-grow

Values: number

Default: 0

Applies to: flex items

Inherits: no

THE MARKUP

```
<div id="container"> <div class="box box1">1</div> <div class="box  
box2">2</div> <div class="box box3">3</div> <div class="box box4">4</div>  
<div class="box box5">5</div></div>
```

THE STYLES

```
.box { ... flex: 1 1 auto;}
```

flex: 0 1 auto; (prevents expansion)



flex: 1 1 auto; (allows expansion)



- اگر مقدار صفر باشد
یعنی اجازه
گسترش ندارد و در
صورتی که یک باشد
اجازه گسترش دارد
- در مثال مقابل اجازه
گسترش دارند

Flex, flex-basis, flex-shrink

```
.box4 { flex: 3 1 auto;}
```



flex: 3 1 auto;

در مثال بالا عددی که برای گسترش نوشته شده ۳ است یعنی در صورت گسترش ۳ برابر بقیه گسترش پیدا خواهد نمود

flex-shrink

Values: *number*

Default: 1

Applies to: flex items

Inherits: no

اجازه کاهش و مقدار کاهش را مشخص می‌کند

مقدار اولیه آیتم را مشخص می‌کند

flex-basis

Values: *length* | *percentage* | *content* | *auto*

Default: *auto*

Applies to: flex items

Inherits: no

Flex , order

`flex: 0 1 100px;`



- وقتی عنصر نگهدارنده عریض است ولی چون مقدار `flex-grow` برابر با صفر است آیتم ها گسترش پیدا نمی کنند. عرض هر آیتم ۱۰۰ پیکسل است
- در عوض زمانیکه عنصر نگهدارنده کوچک باشد عرض آیتمها کاهش پیدا می کند .

order

Values: *integer*

Default: 0

Applies to: flex items and absolutely positioned children of flex containers

Inherits: no

ترتیب آیتمها را مشخص می کند
و بر روی آیتمها اعمال می شود

`.box3 { order: 1;}`



`order: 0`
(default)

`order: 0`
(default)

`order: 0`
(default)

`order: 0`
(default)

`order: 1`