# CSS Flexbox

CSS Flexbox is a mechanism provided by CSS using which we can arrange elements on the UI.

It is a very simple, efficient and flexible way to create layouts, arrange elements, align them horizontally or vertically, distribute them in the space available etc.

Flexbox was not always the part of core css, and prior to this in order to do any arrangements, developers had to use floats, tables, positions etc to create any complex layout.

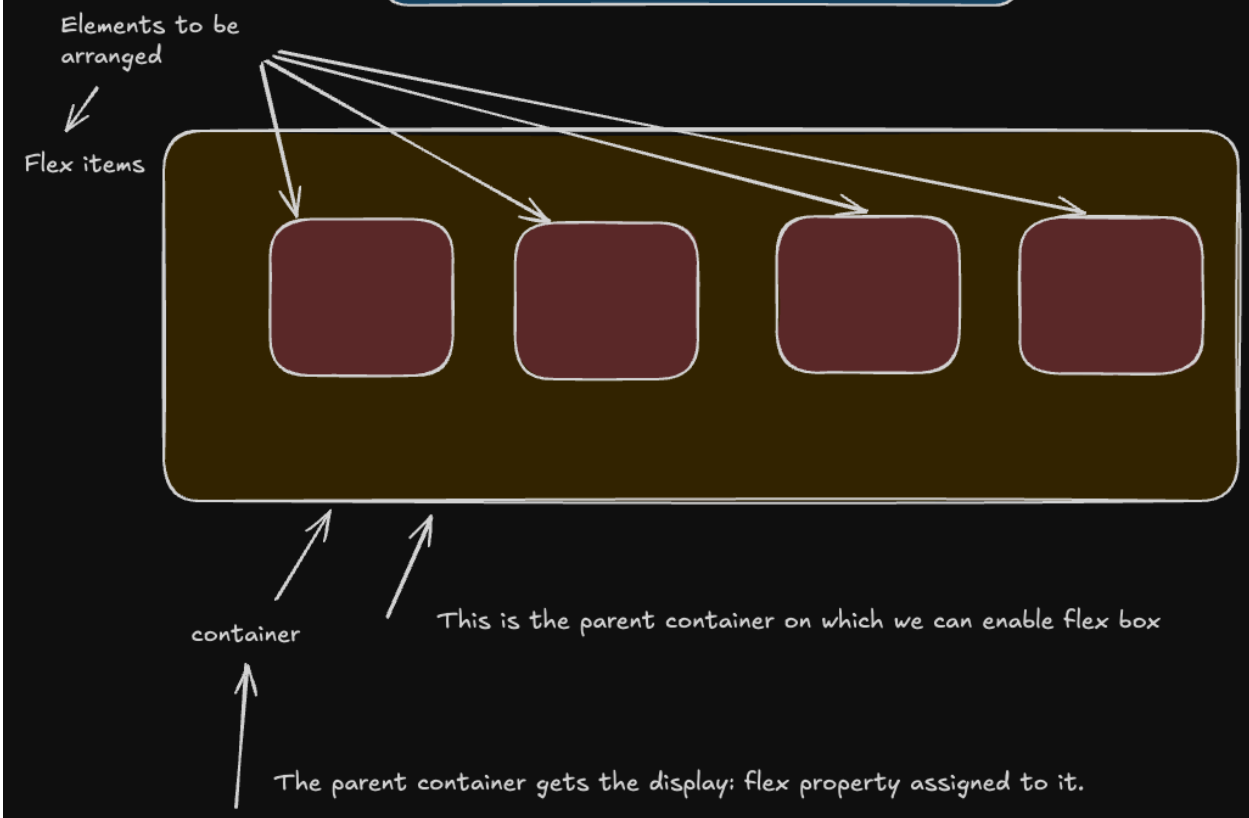
To enable flex box, we use the display property and give it a value flex. By giving display:flex, we enable flexbox for the layout.

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

This code pointer enables flexbox.

---

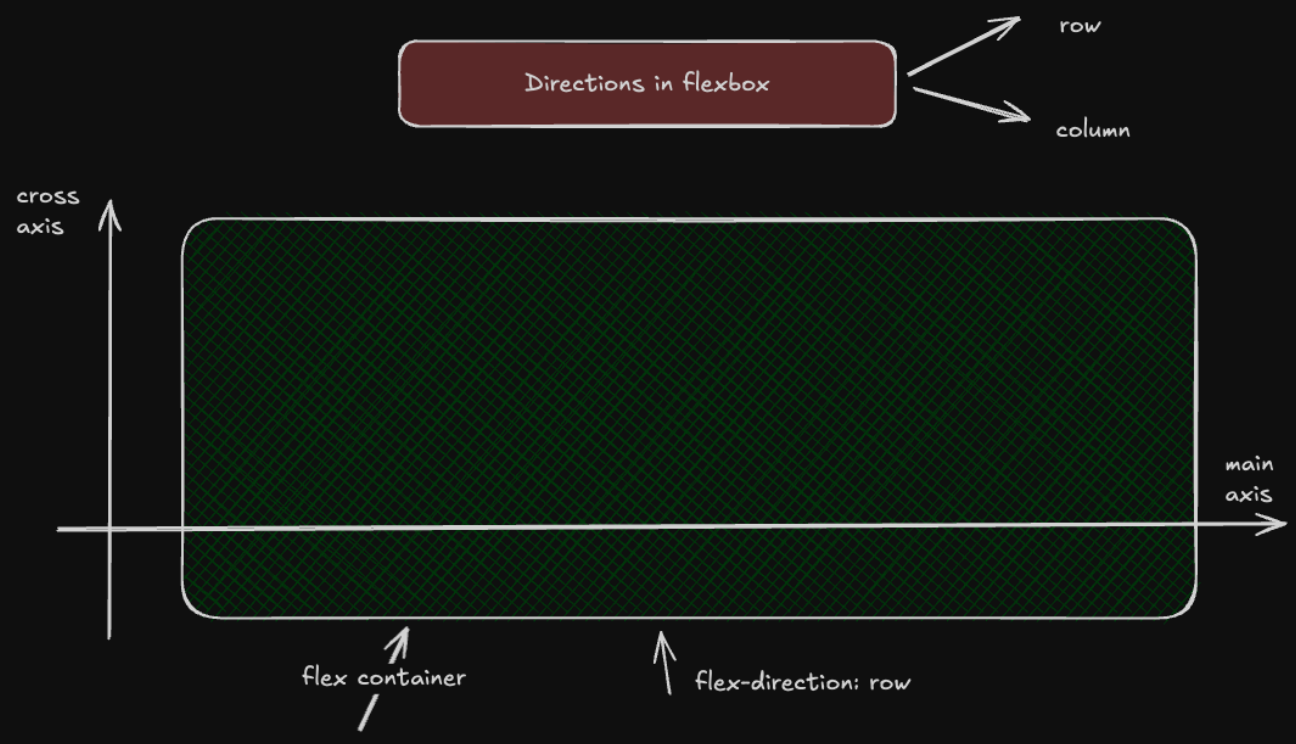
## Flex containers & Flex items



Flex container is a parent element having child elements known as flex items.

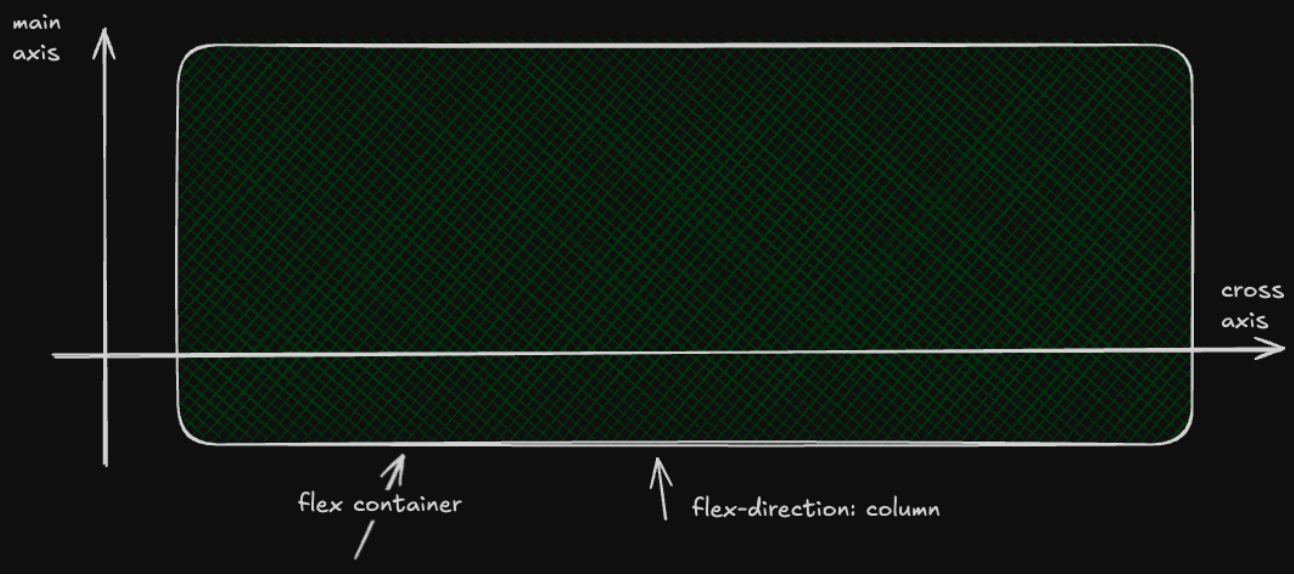
Flex items are arranges and aligned using flexbox based properties on the container.

The moment we give display: flex to the container, immediately the container flex is activated & all the direct elements are classified as the flex items.




In direction set as row, elements/items are arranged in the same row. With direction set as column, items are arranges in the same column.

To control the direction of items, we can set flex-direction property on the container.



#main axis: the main axis is the primary axis along which the elements will be arranged in the container.

flex-direction: row ----> main axis is x-axis
flex-direction: column ----> main axis is y-axis

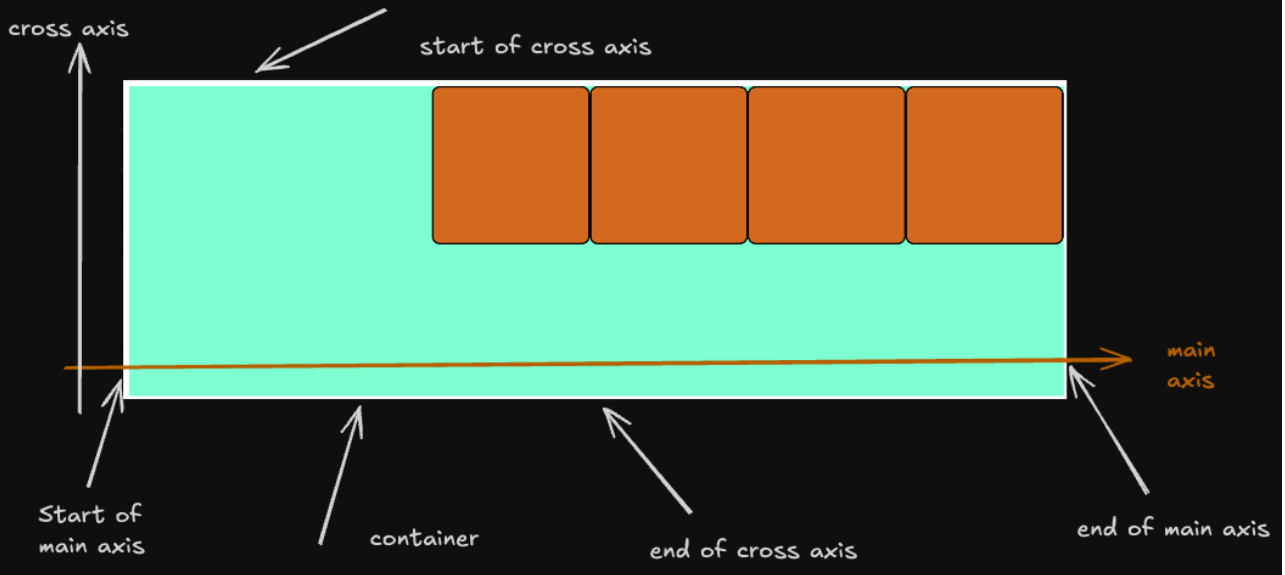
# cross axis: the cross axis is the axis perpendicular to the main axis.
If main axis is x, then cross axis is y and vice-a-versa

## Justify content

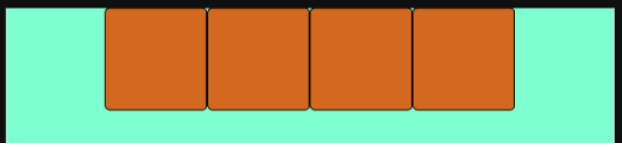
- This property is used to distribute the available space of main axis among the flex items.

direction: row ---> main axis : x-axis ----> justify content will control row arrangement

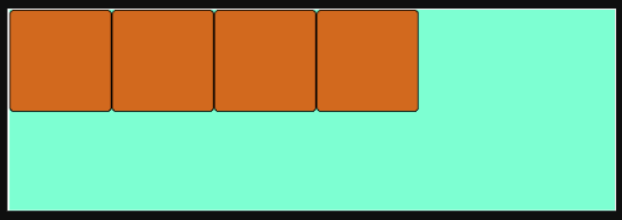
direction: column ---> main axis: y-axis ----> justify content will control column arrangement.

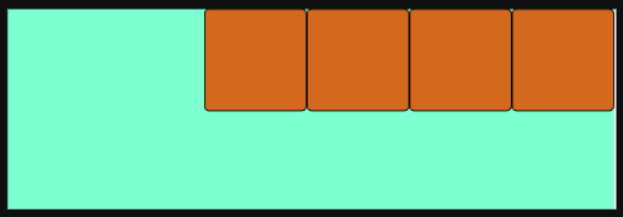

1. center: this helps align the items on the center of the main axis.



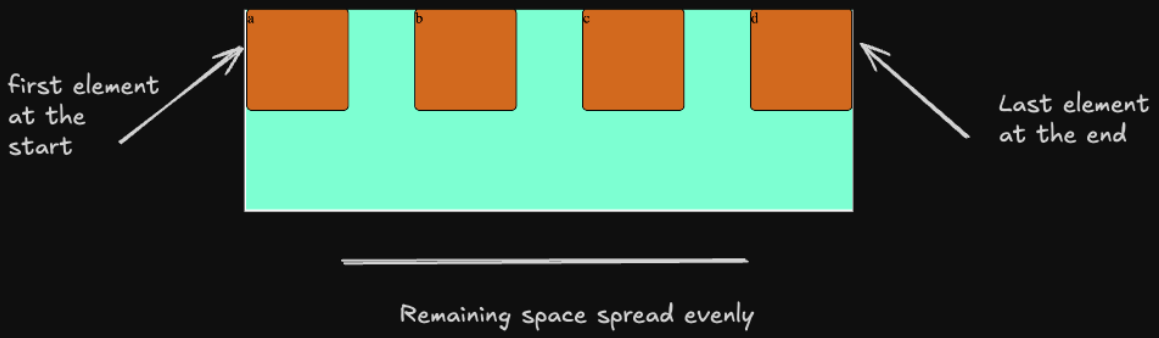
2. start: this helps align the items on the start of the main axis



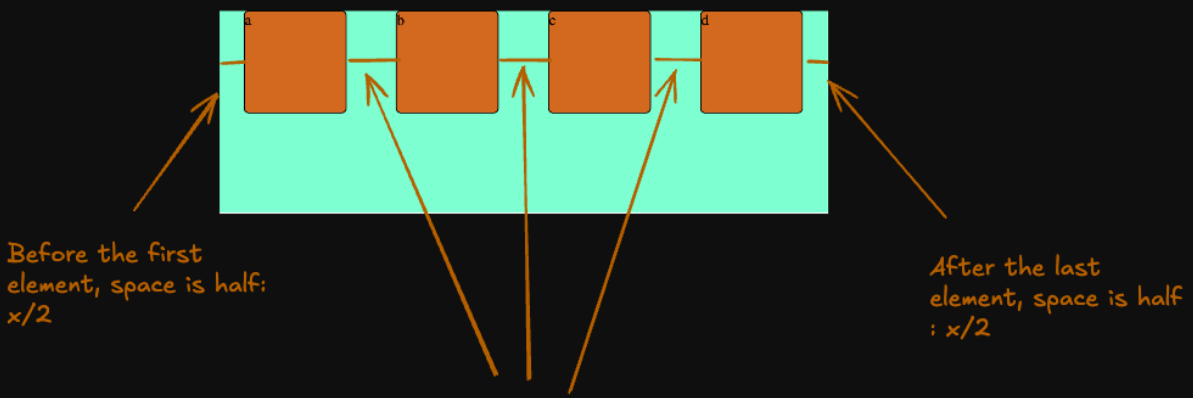
3. end: this helps align the items on the end of the main axis. Element order is still right but they are aligned towards the end of main axis , or we can say, they take space from the end.



4. space-between: This will make your items go as far as possible. It takes the first item in the container and puts it at the absolute start of the main axis, then takes the last element, and puts it at the absolute end of main axis, and then whatever space is remaining it distributes that space evenly among the middle elements.

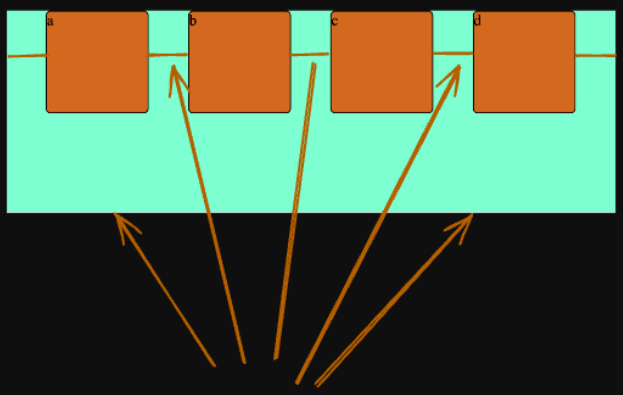



5. space-around: This is going to evenly divide the space between items.



same spacing: x

6. space-evenly: this will evenly divide the space among all the element, i.e. even with space before start and space after end.



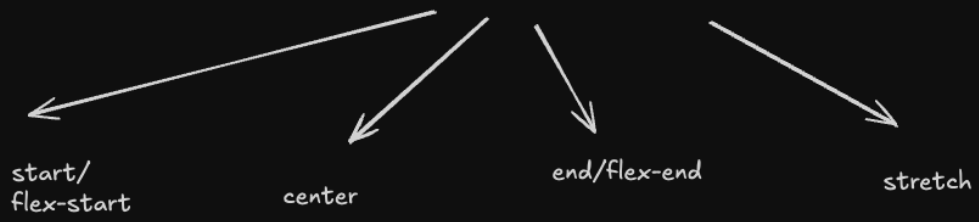

All the space have same value: x

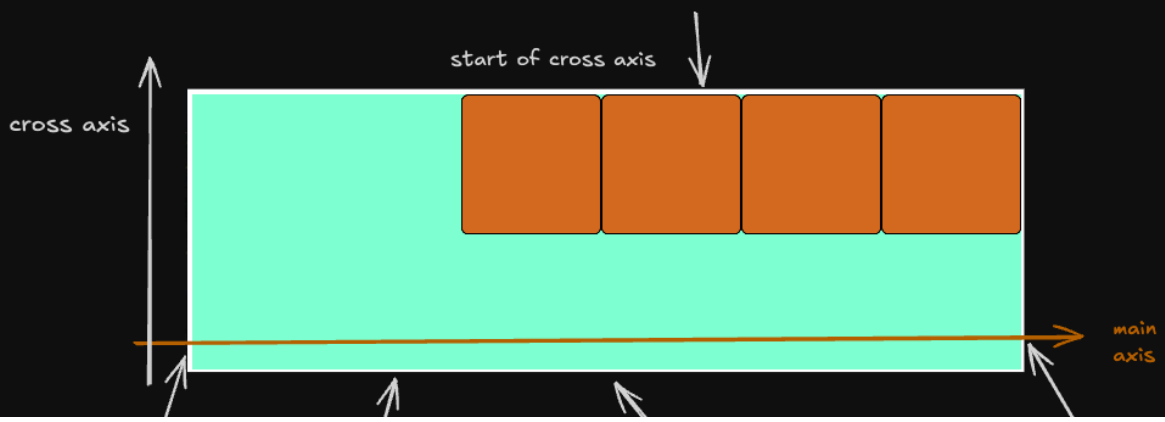
## Align Items Property

The align-items property distributes the available space of cross axis between the flex items

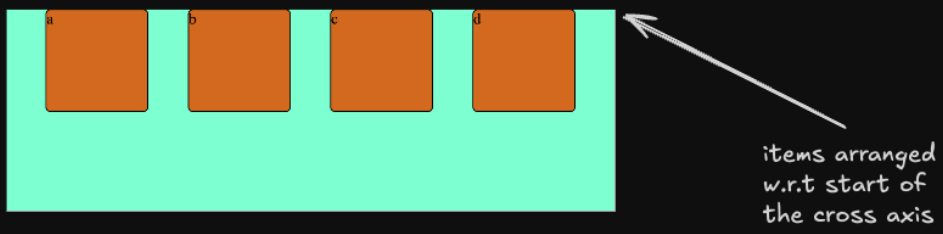
direction: row ---> cross axis y-axis ---> align-items control the vertical arrangement

direction: column ---> cross axis: x-axis ---> align-items control the horizontal arrangement

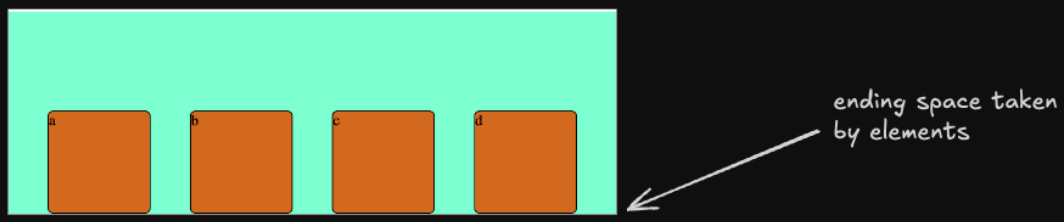


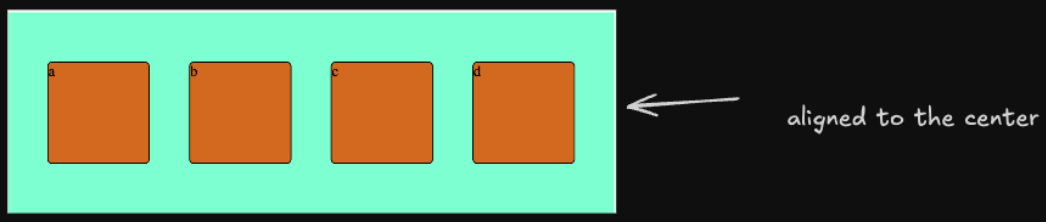

1. start: this will keep the elements take the start of the cross axis space. This is the default value as well.



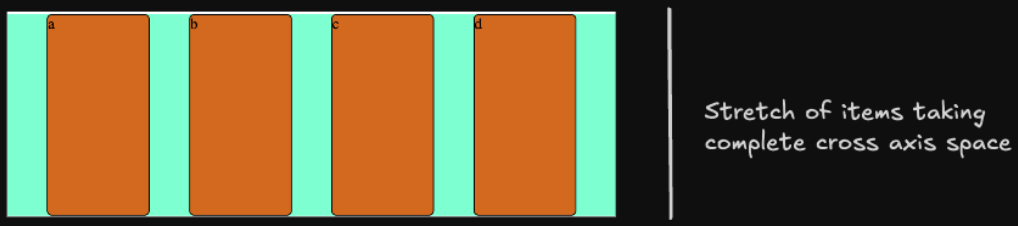

2. end: this will keep the elements take the end of cross axis space.




3. center: this will align the elements to the center of the cross axis

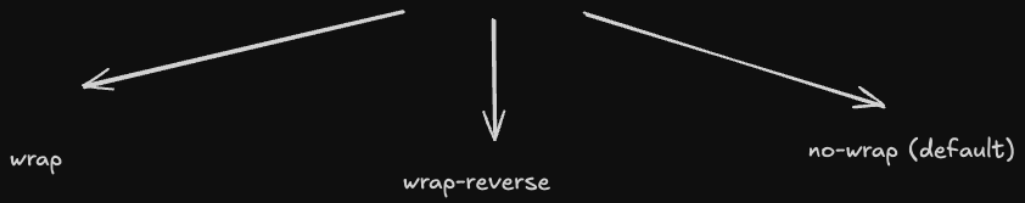



4. stretch: this value makes the items stretch to fill the complete container cross axis space.



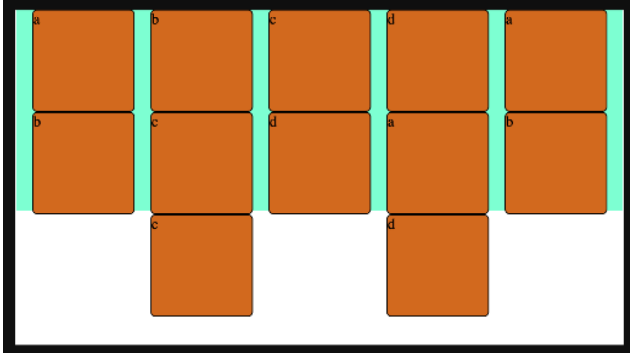
## Flex Wap

The flex wrap value allows flex items to wrap in multiple lines. By default all the flex items are squeezed in a single horizontal direction (row).

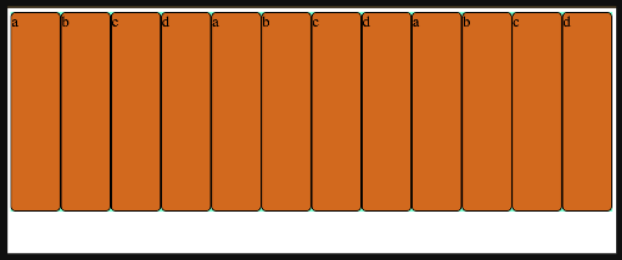



# wrap: the wrap value allows flex items to be wrapped in multiple lines when their size is too big for the container.

# wrap-reverse: this will wrap also but in reverse order



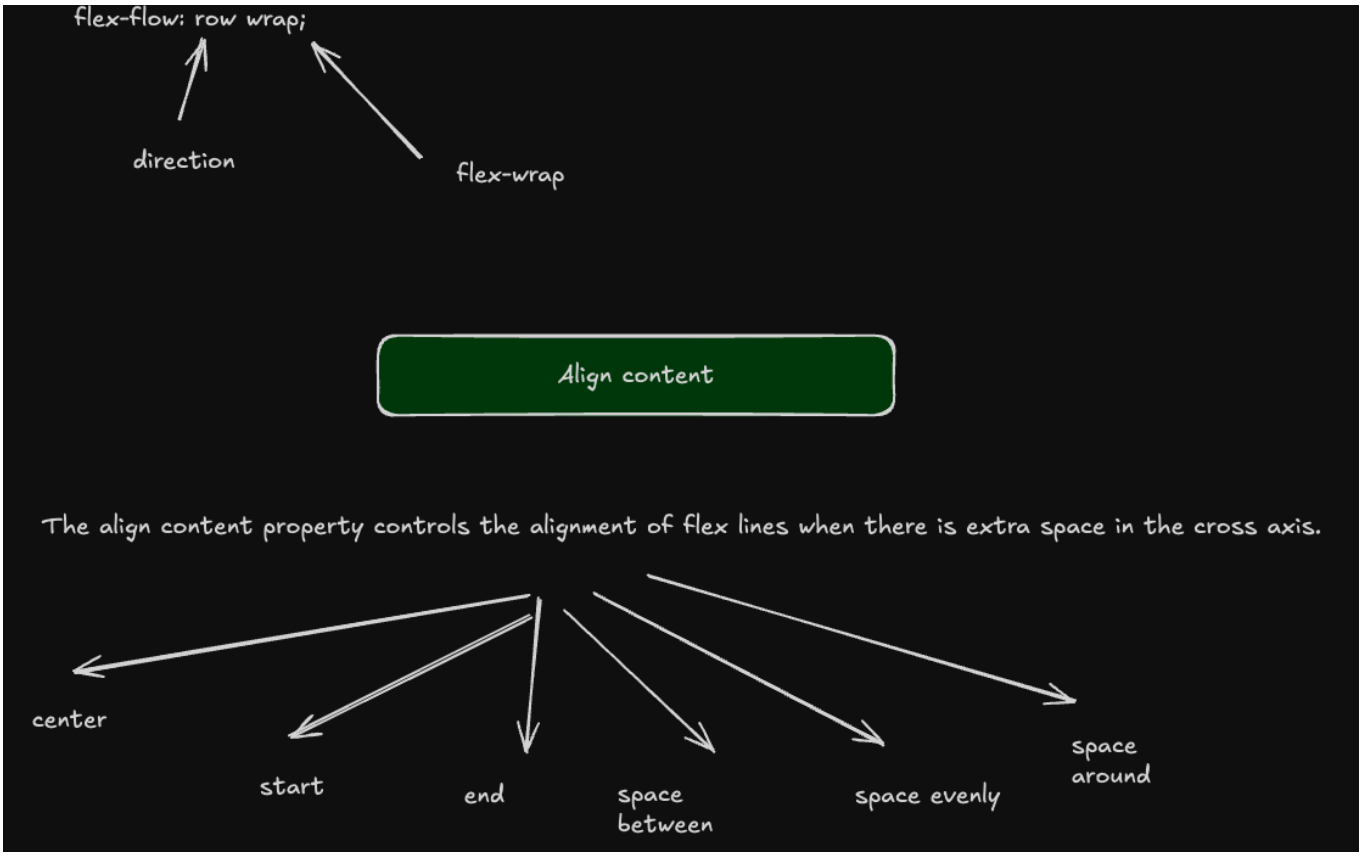

flex-wrap: wrap




flex-wrap: nowrap

## Flex flow

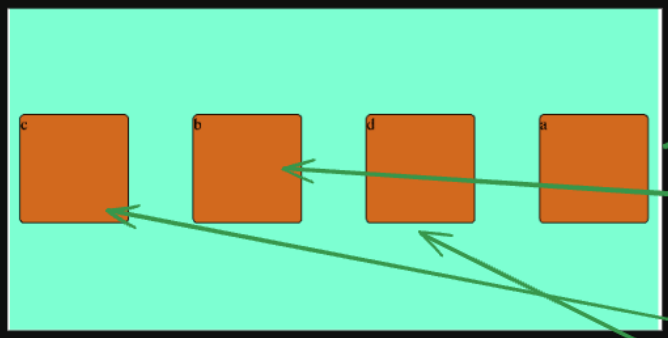
Shorthand property to specify both flex-direction and flex-wrap.


flex-flow: row wrap;
direction
flex-wrap
Align content
The align content property controls the alignment of flex lines when there is extra space in the cross axis.
center
start
end
space between
space evenly
space around

## Order Property

The flex items appear in the same order as they are present in the HTML document by default.

If we want to customise this ordering then flex order property is very useful. This can specify order of the flex items in the flex container.




```
22
23  #a {
24      order: 4;
25  }
26
27  #b{
28      order: 2;
29  }
30
31
32  #c {
33      order: 1;
34  }
35
36  #d {
37      order: 3;
38  }
39
40  /* c b d a */
```

Default ordering was

a b c d

New ordering of the
flex items inside the container

The flex order property can take positive and negative values, with lower values placing the flex items first in the container.

## Flex grow property

The flex grow property determines how much flex items expand relative to the other items in the flex container.

Its a unit less value representing proportion of the remaining space in the container

## Flex shrink property

This property determines how much flex items shrink relative to the other items when there is not enough space in the flex container.