

Flexbox

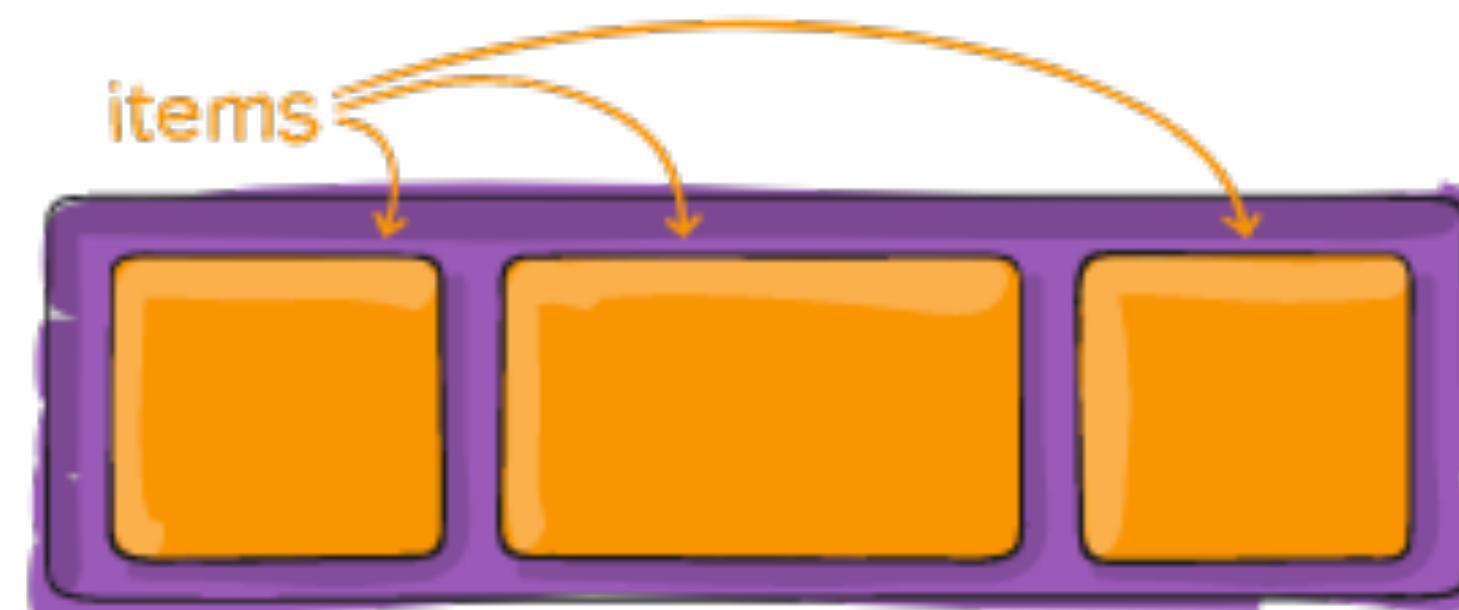
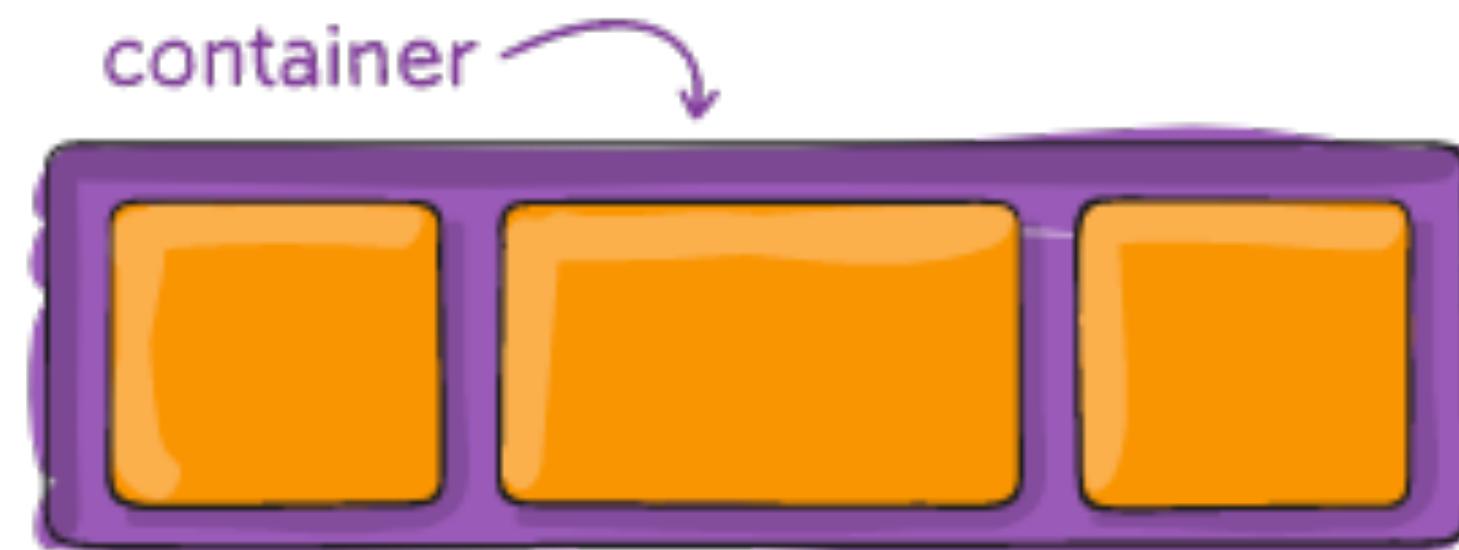
WEEK 5

Flexbox

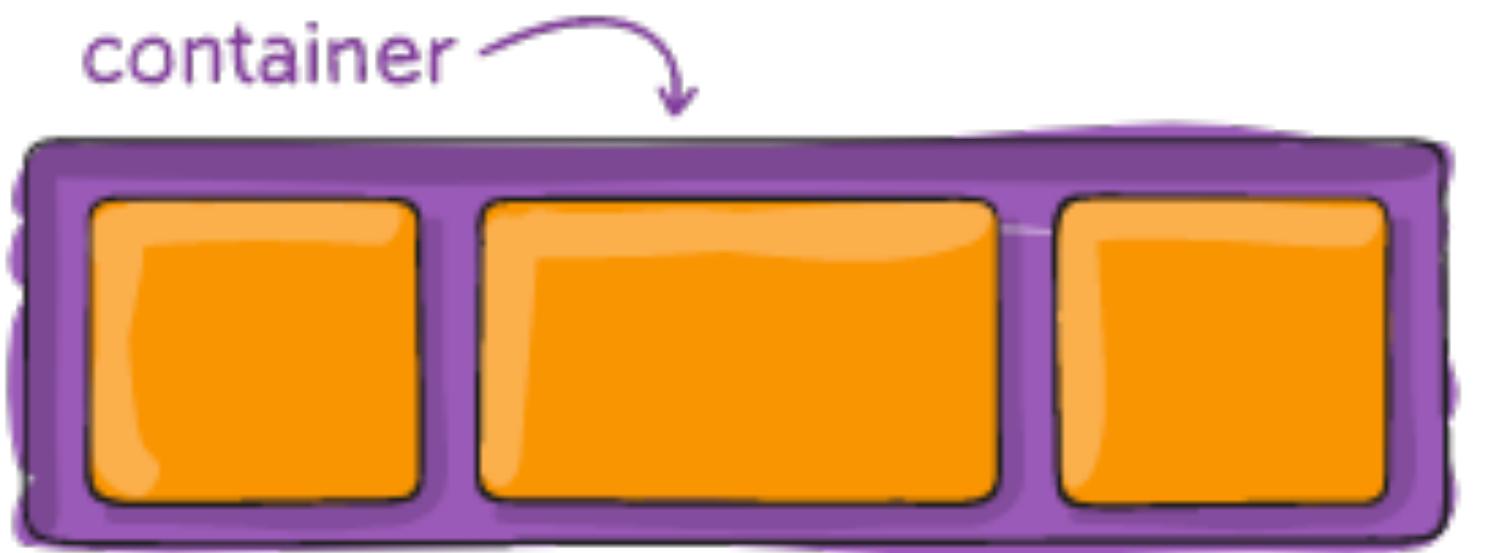
- came after floats and is widely used
- specific CSS properties
- easier alignment

[[CSS-Tricks Guide to Flexbox >>](#)]

[[W3 Schools Reference >>](#)]

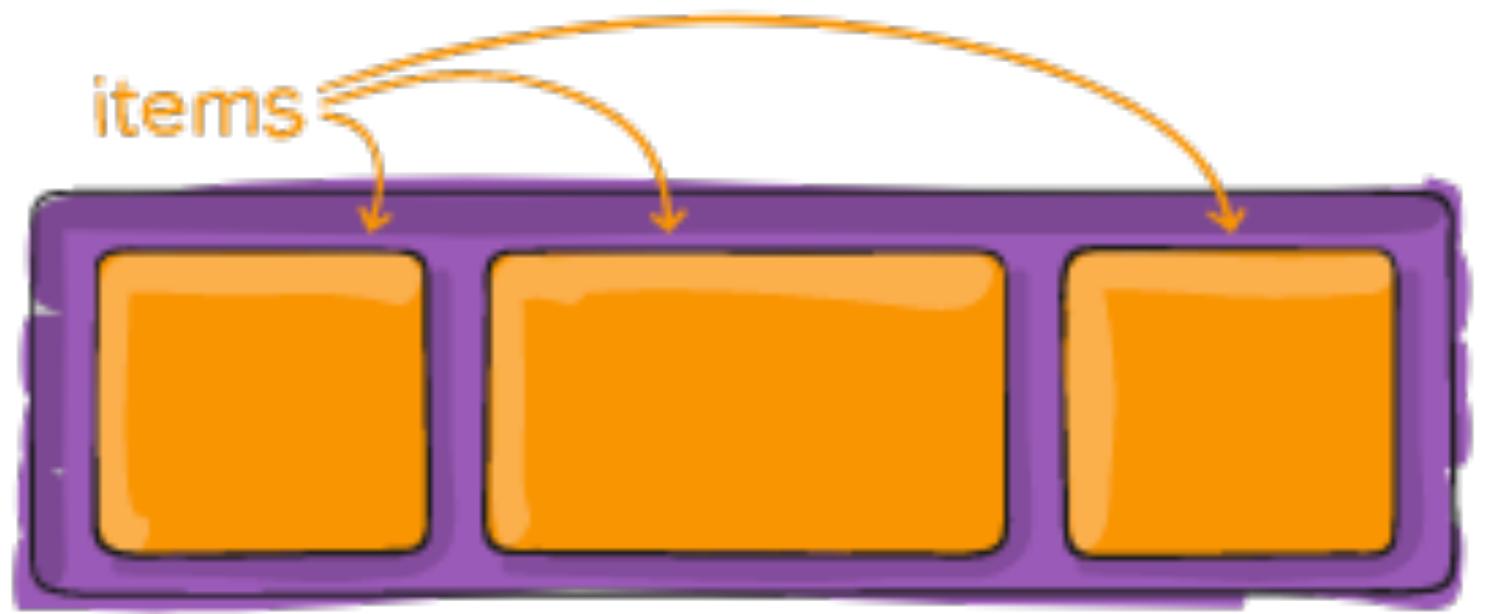


FLEXBOX



```
.flex-container {  
  display: flex;  
  /* styles for the big box */  
}
```

FLEXBOX



```
.flex-container > div {  
  /* once you define a flex-container,  
   the browser knows that the direct  
   children of that container are flex  
   items */  
}
```

FLEXBOX CONTAINER

{flex-direction}

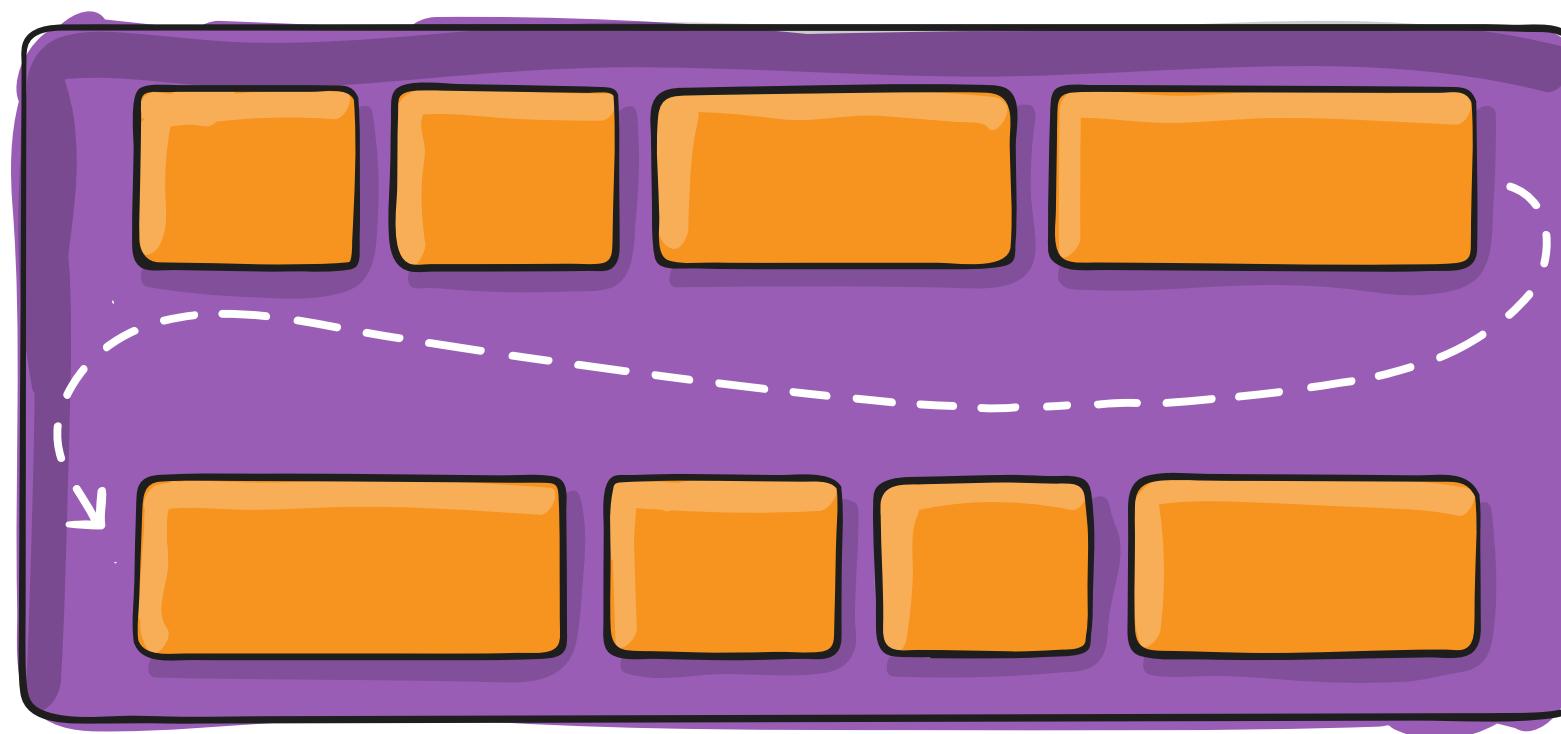


```
.flex-container {  
  display: flex;  
  flex-direction: row;  
}
```

[W3 Reference >>]

FLEXBOX CONTAINER

{flex-wrap}



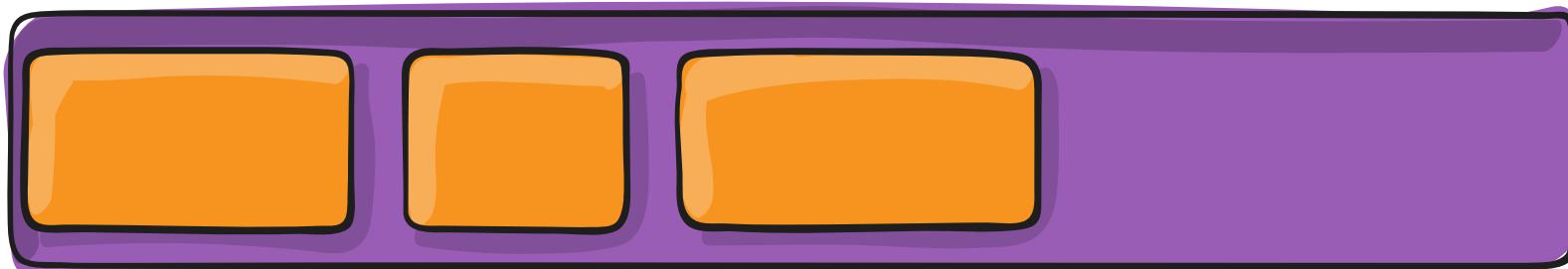
```
.flex-container {  
  display: flex;  
  flex-wrap: wrap;  
}
```

[[W3 Reference >>](#)]

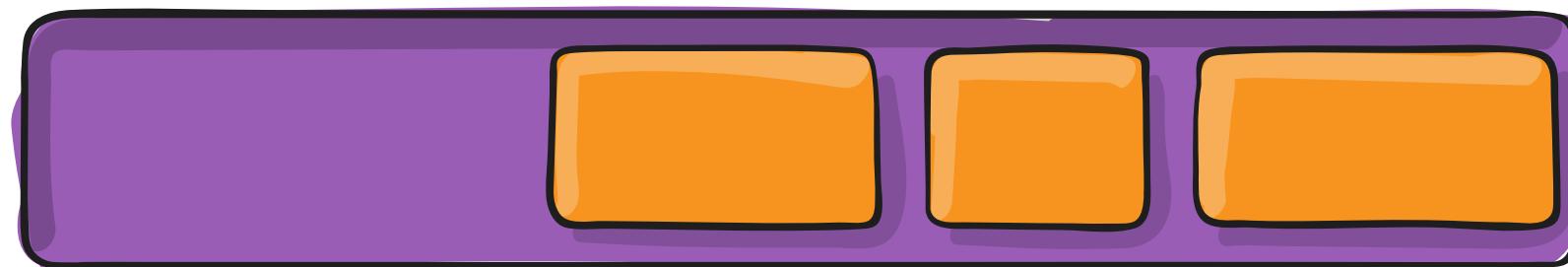
FLEXBOX CONTAINER

{justify-content}

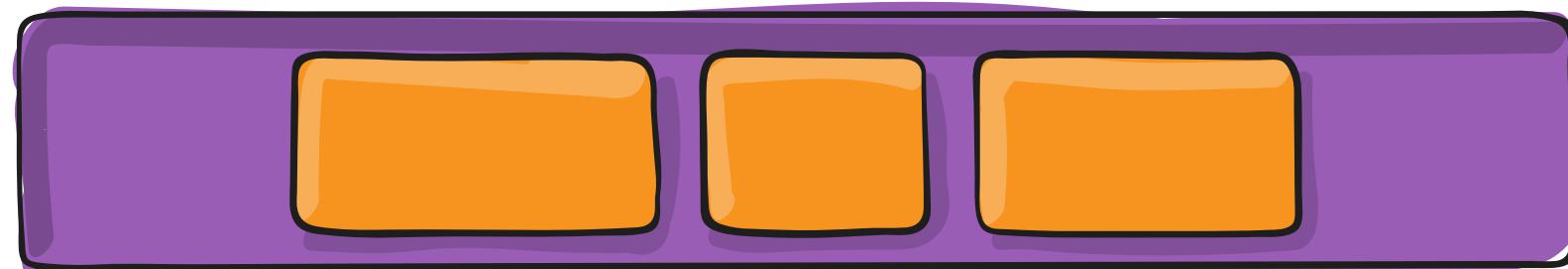
flex-start



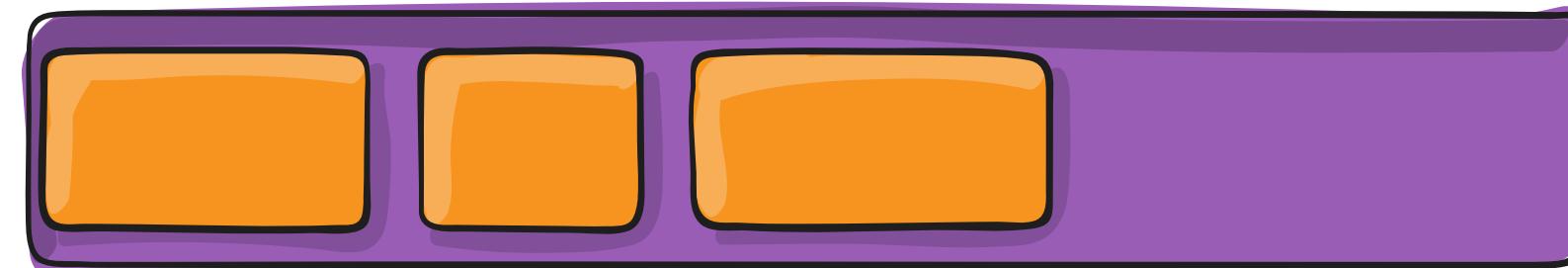
flex-end



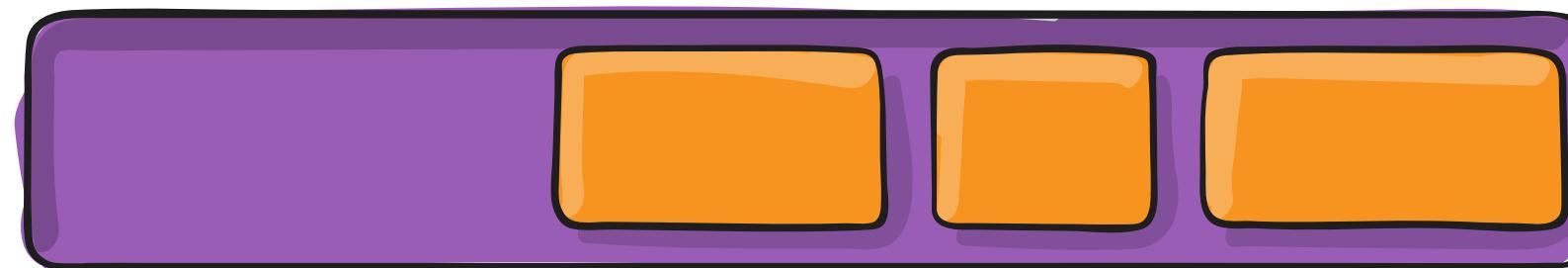
center



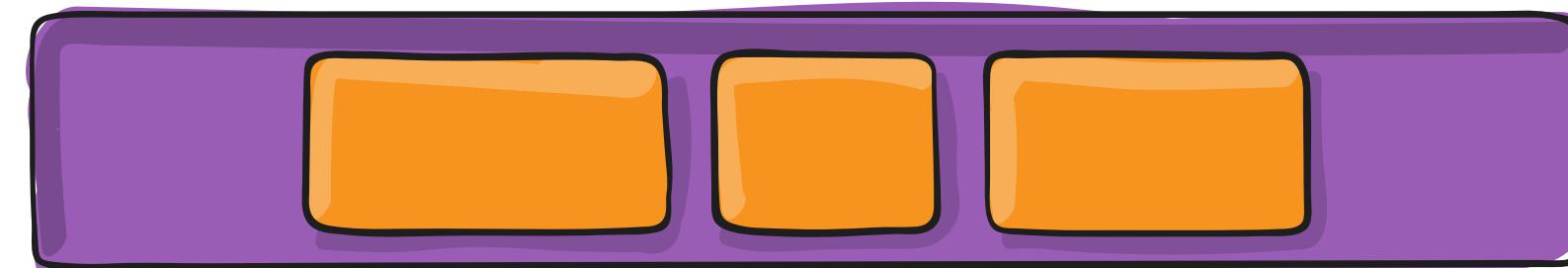
flex-start



flex-end



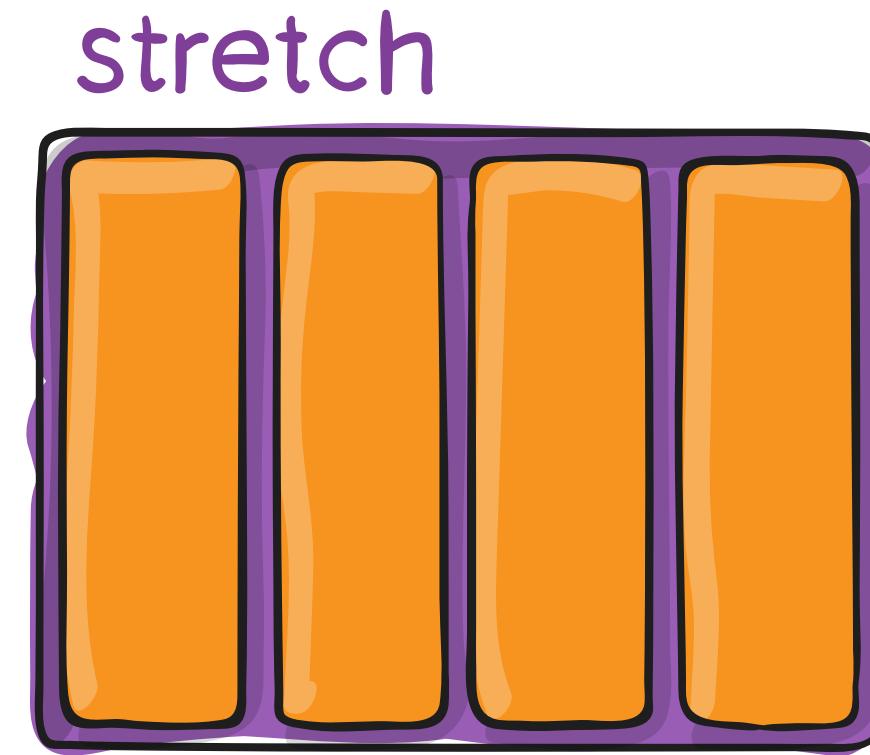
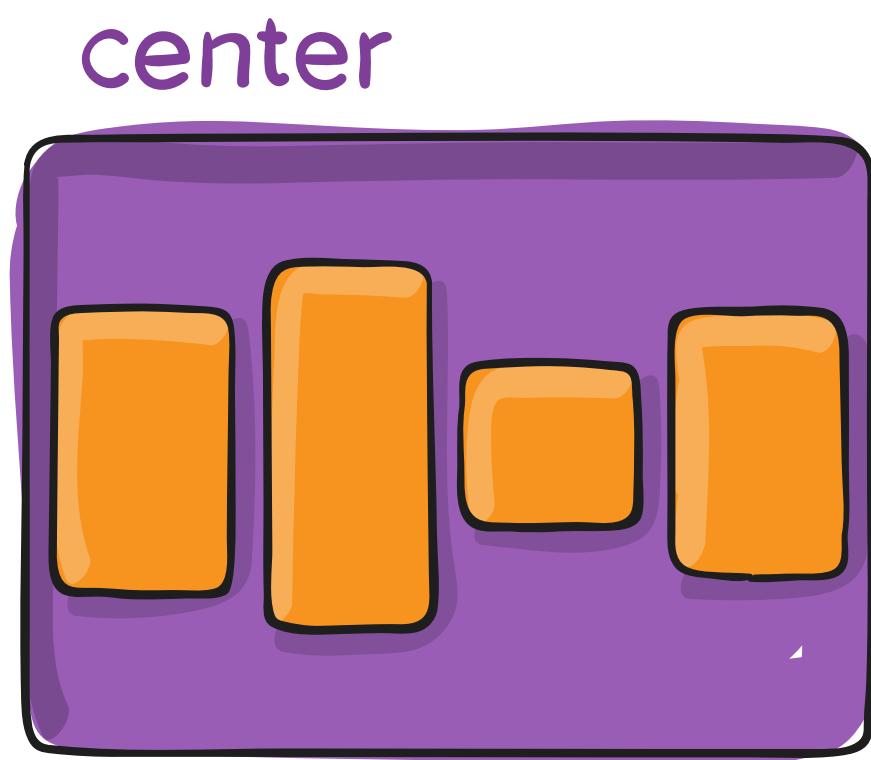
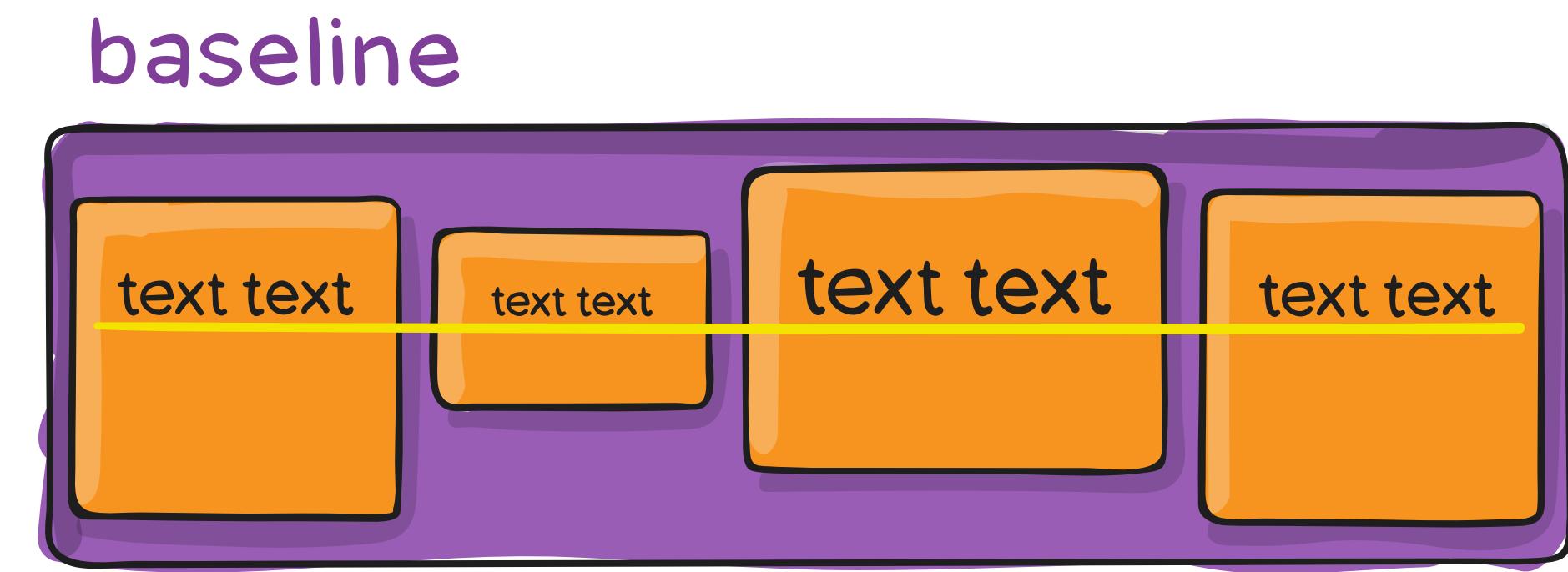
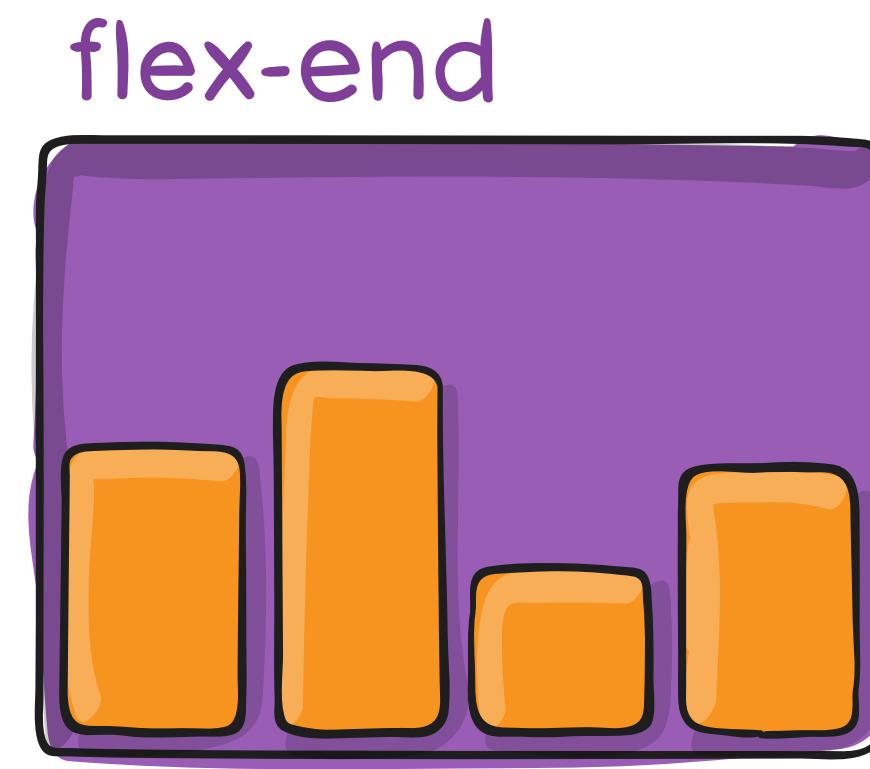
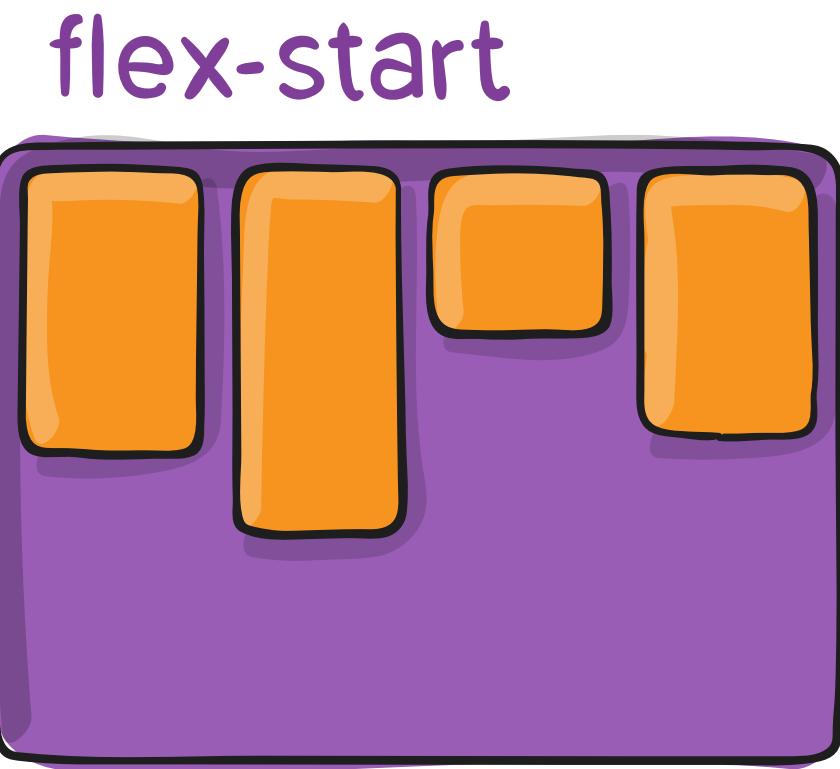
center



[W3 Reference >>]

FLEXBOX CONTAINER

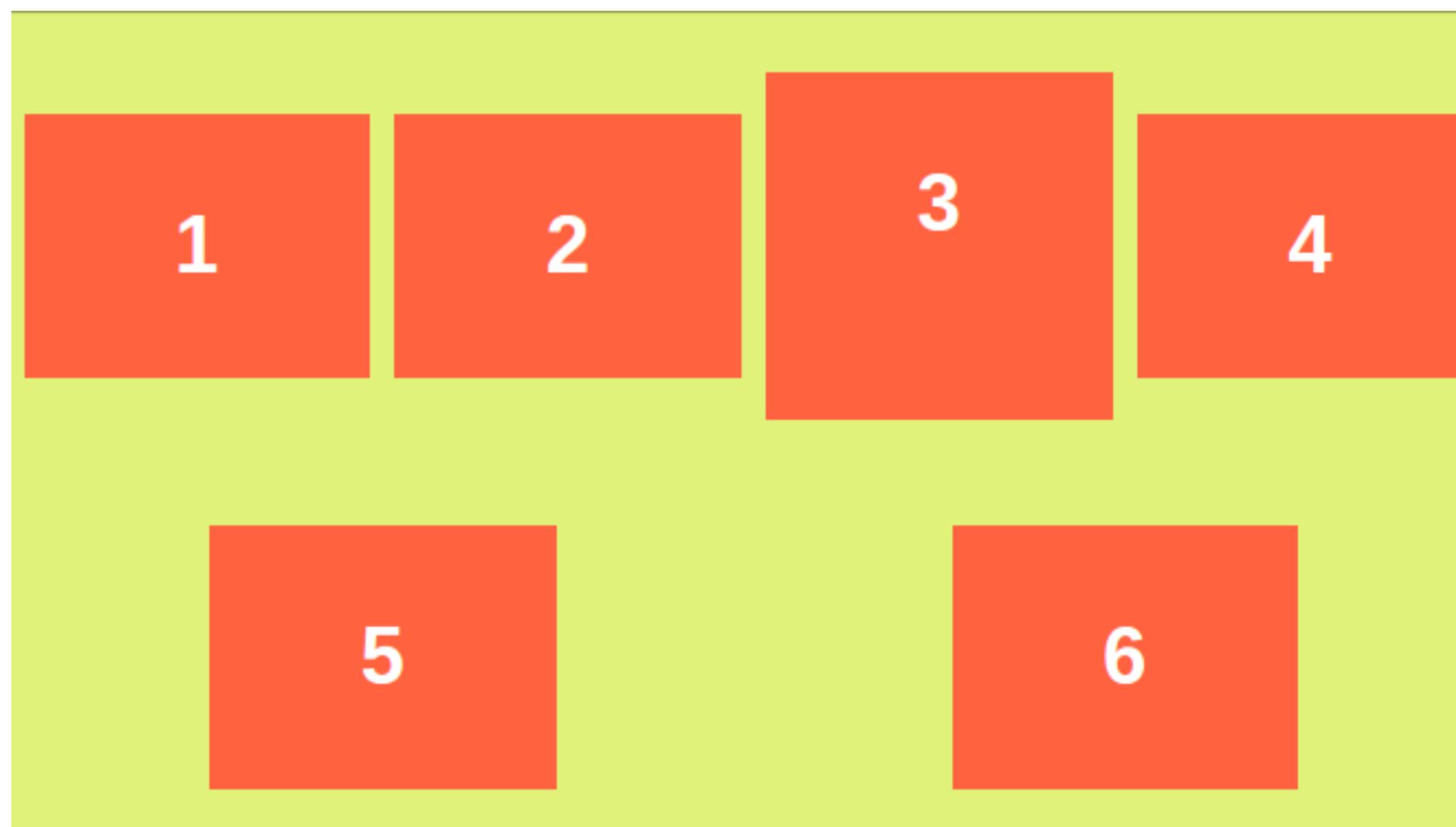
{align-items}



FLEXBOX CONTAINER

{align-items} vs {align-content}

{align-content} is for multi line flexible boxes.
It has no effect when items are in a single line.
It aligns the whole structure according to its value.



align-content: space-around;
align-items:center

FLEXBOX CONTAINER

**TO CENTER-ALIGN in flex container
classes horizontally and vertically**

```
.flex-container{  
    justify-content: center;  
    align-items: center;  
}
```

FLEXBOX CONTAINER

FLEX CONTAINER

the flex container needs to have
`{display:flex;}`

flex-direction:

column, column-reverse, row, row-reverse;

flex-wrap:

wrap, nowrap wrap-reverse;

flex-flow:

(sets both flex-wrap and flex-direction)

justify-content (aligns horizontally):

center, flex-start, flex-end, space-around;

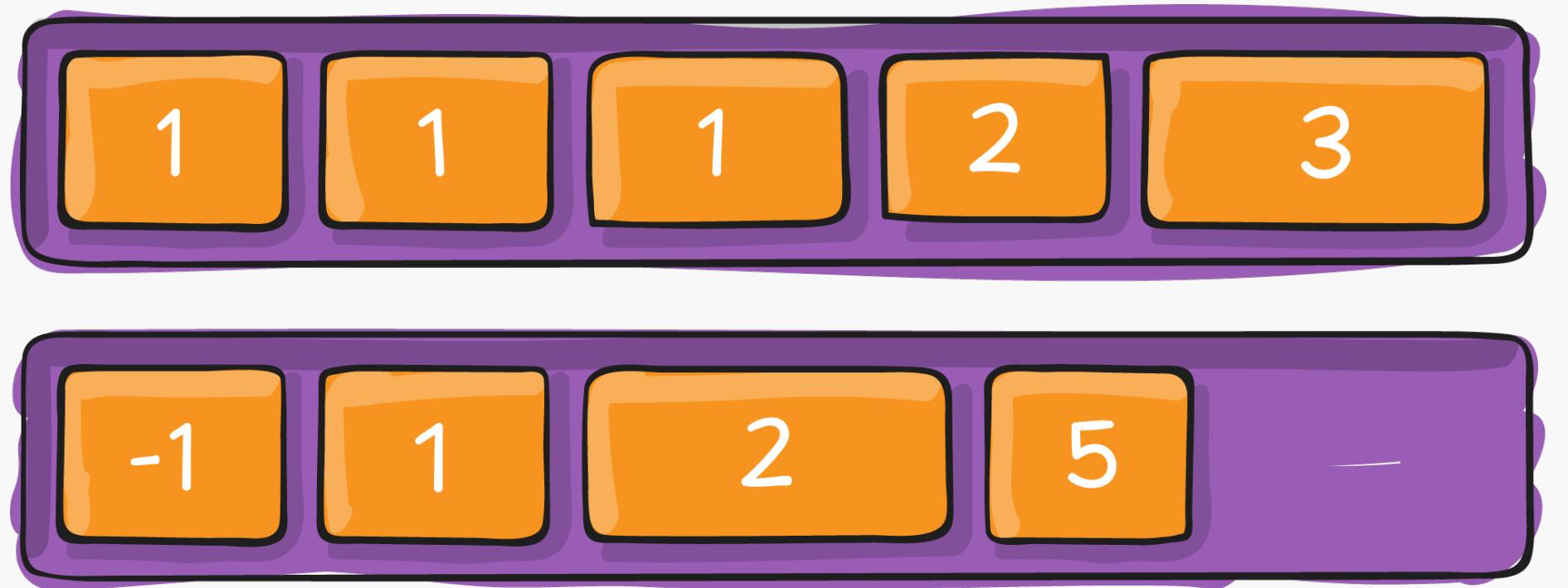
align-items (aligns flex items vertically)
center, flex-start, flex-end, stretch, baseline;

align-content

(aligning flex lines, so this works with wrapping the content): space-between, space-around, stretch, center, flex-start, flex-end,

FLEXBOX ITEMS

{order}

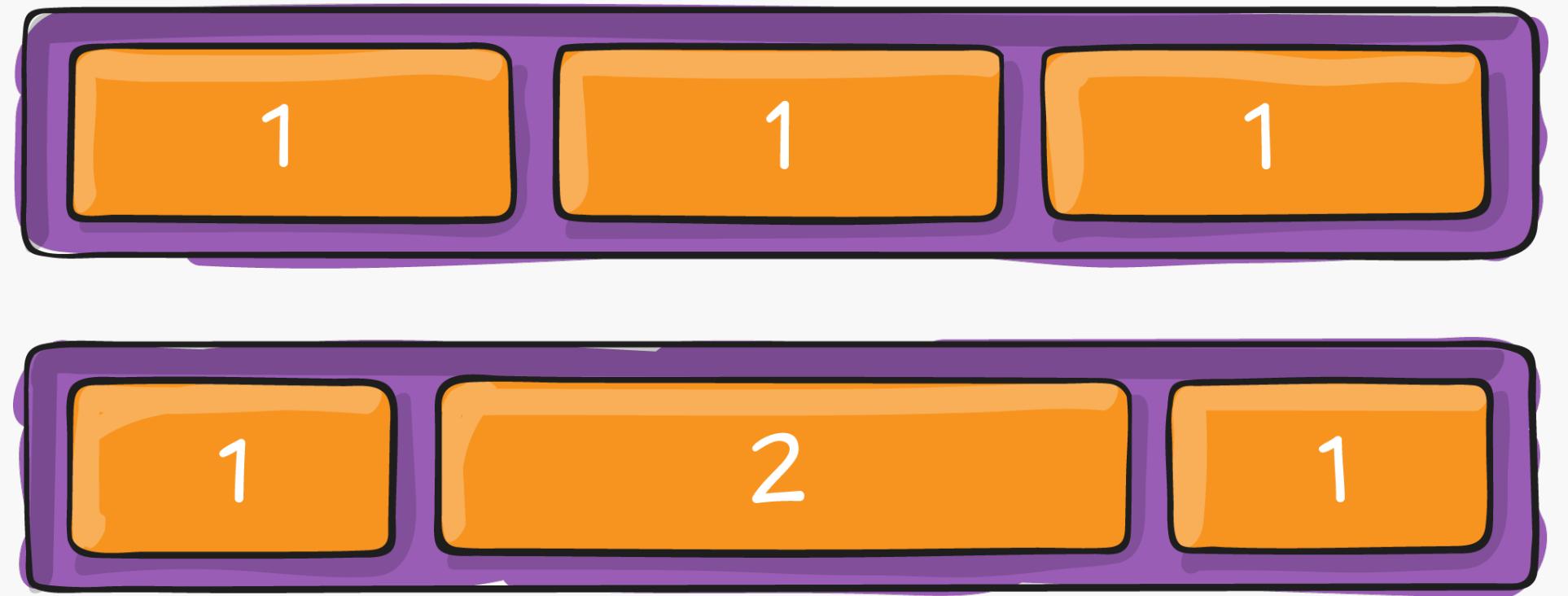


```
.item {  
  order: 5; /* default is 0 */  
}
```

[[W3 Example >>](#)]

FLEXBOX ITEMS

{flex-grow}

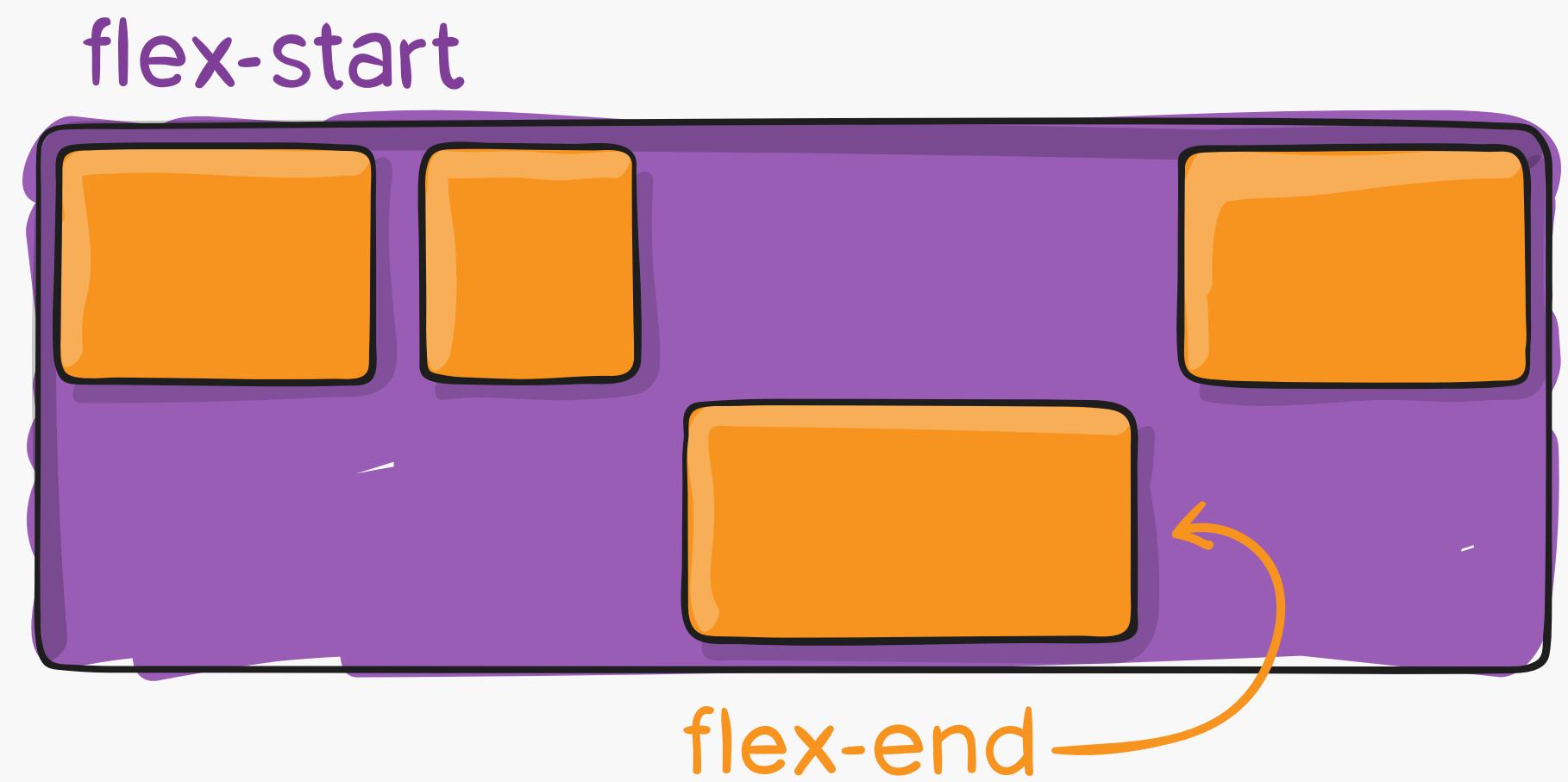


```
.item {  
  flex-grow: 8;  
}
```

[[W3 Example >>](#)]

FLEXBOX ITEMS

{align-self}



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

FLEXBOX ITEMS

FLEX CONTAINER

Do not have to define them specifically, browsers will know that direct children are flex items

order (*reorder the item*):

{order: 3;}

/* reorders the item to be 3rd in the list*/

flex-grow

(*the higher the number, >> grows faster*):

{flex-grow:8;}

flex-shrink:

{flex-shrink:0;}

/* doesn't shrink at all, stays the same*/

flex-basis (*sets initial width of an item*):

{flex-basis:200px;}

flex (shorthand for flex-grow, flex-shrink, flex-basis)

{flex: 0 0 200px;}

align-self (*overrides align-items property*):

{align-self:center}

BROWSER SUPPORT

CSS browser prefixes, are a way for browser makers to add support for new CSS features before those features are fully supported in all browsers

[AUTOPREFIXER >>]

```
.example {  
    display: flex;  
    transition: all .5s;  
    background: linear-gradient(to bottom,  
white, black);  
}
```

css

```
.example {  
    display: -webkit-box;  
    display: -ms-flexbox;  
    display: flex;  
    -webkit-transition: all .5s;  
    -o-transition: all .5s;  
    transition: all .5s;  
    background: -webkit-gradient(linear, left  
top, left bottom, from(white), to(black));  
    background: -webkit-linear-gradient(top,
```

BROWSER SUPPORT

CSS

```
.example {  
    display: -webkit-box;  
    display: -ms-flexbox;  
    display: flex;  
    -webkit-transition: all .5s;  
    -o-transition: all .5s;  
    transition: all .5s;  
    background: -webkit-gradient(linear, left  
top, left bottom, from(white), to(black));  
    background: -webkit-linear-gradient(top,
```

