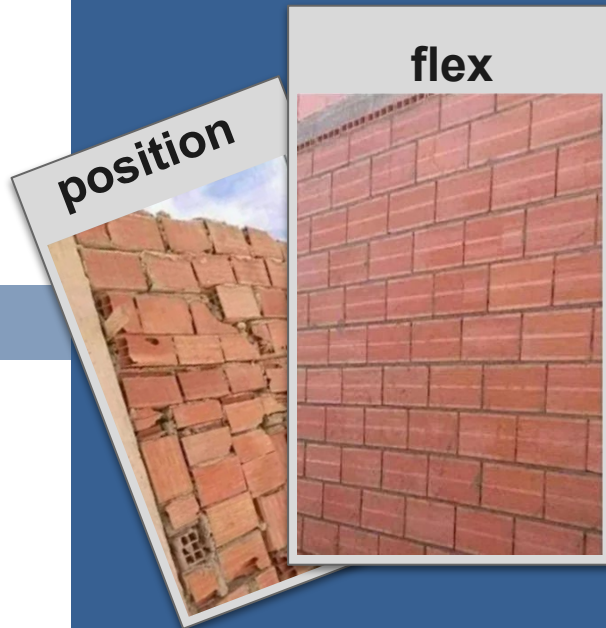


CSS

FlexBox



FlexBox

```
element.style {  
  display: flex;  
}
```

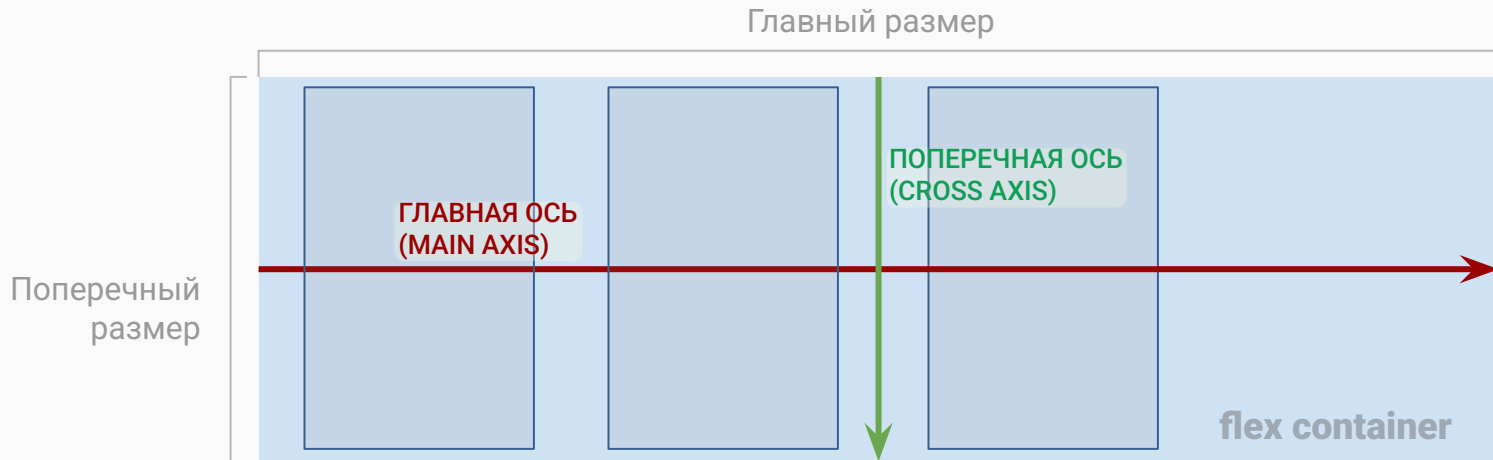
flex

inline-flex

Flexbox (Flexible Box Layout) - это модуль CSS3, который позволяет создавать гибкие контейнеры и управлять расположением элементов внутри этих контейнеров.

С помощью Flexbox можно легко выравнивать элементы по горизонтали и вертикали, управлять порядком элементов и их размерами.

flex контейнер



Значение **display: flex** превращает блок во flex-контейнер.

Flex контейнер имеет 2 оси: главную (*main axis*) и поперечную (*cross axis*)

По умолчанию все предметы располагаются вдоль главной оси — слева направо.

Свойства flex контейнера

justify-content определяет выравнивание вдоль главной оси



justify-content : flex-start

(по умолчанию): элементы сдвинуты в начало **flex-direction** направления



justify-content : flex-end

элементы сдвинуты ближе к концу **flex-direction** направления



justify-content : start

justify-content : end

аналогично flex-start и flex-end соответственно, но элементы сдвинуты в относительно **writing-mode** направления

justify-content : center

элементы центрированы вдоль линии

justify-content определяет выравнивание вдоль главной оси



justify-content : space-between

элементы равномерно распределены по линии; первый элемент находится в начале строки, последний элемент в конце строки.



justify-content : space-around

элементы равномерно распределены по линии с одинаковым пространством вокруг них.

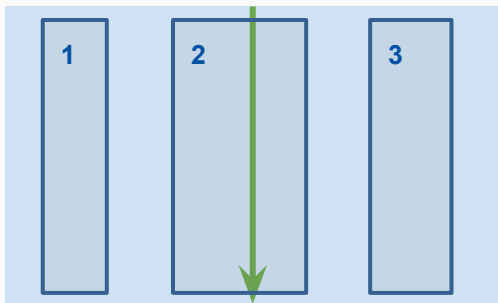


justify-content : space-evenly

элементы распределяются таким образом, чтобы расстояние между любыми двумя элементами (и расстояние до краев) было одинаковым

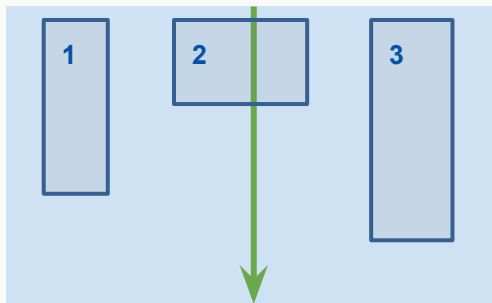
align-items

align-items определяет выравнивание вдоль поперечной оси



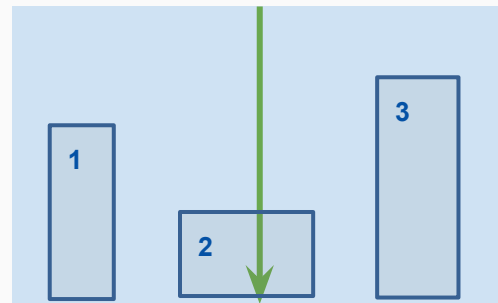
align-items : stretch

(по умолчанию): растягивать, чтобы заполнить контейнер (все еще соблюдаются min-width / max-width)



align-items flex-start

элементы размещаются в начале поперечной оси.

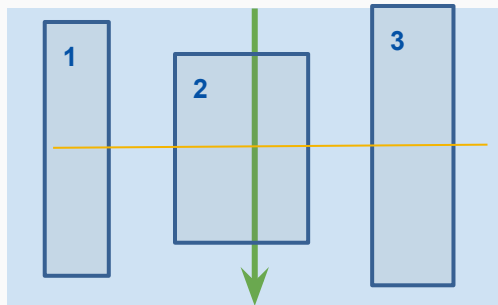


align-items :flex-end

элементы располагаются в конце поперечной оси

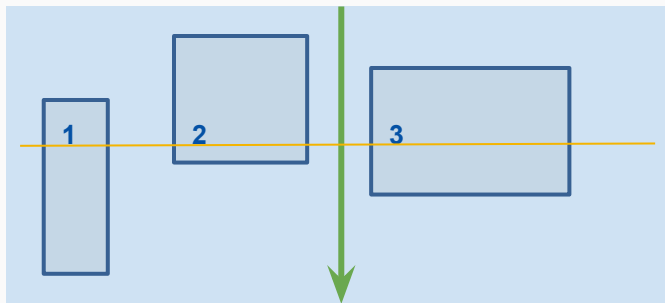
align-items

align-items определяет выравнивание вдоль поперечной оси



align-items : center

элементы центрированы по поперечной оси

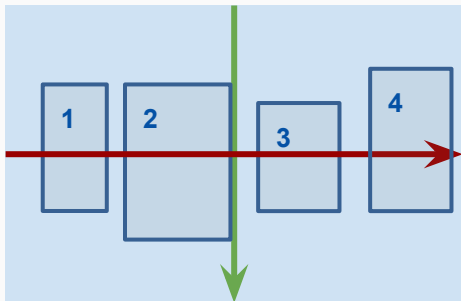


align-items baseline

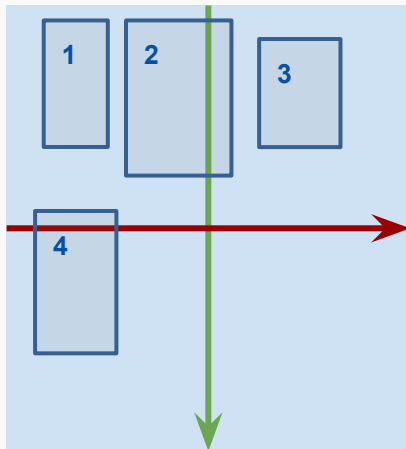
элементы выровнены, по их базовой линии

flex-wrap

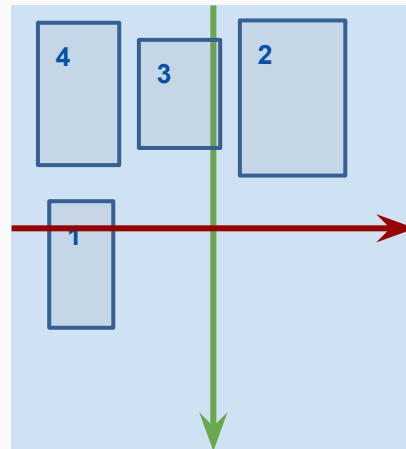
flex-wrap - устанавливает режим переноса элементов на новую строку, по умолчанию гибкие элементы будут пытаться уместиться на одной строке (**nowrap**).



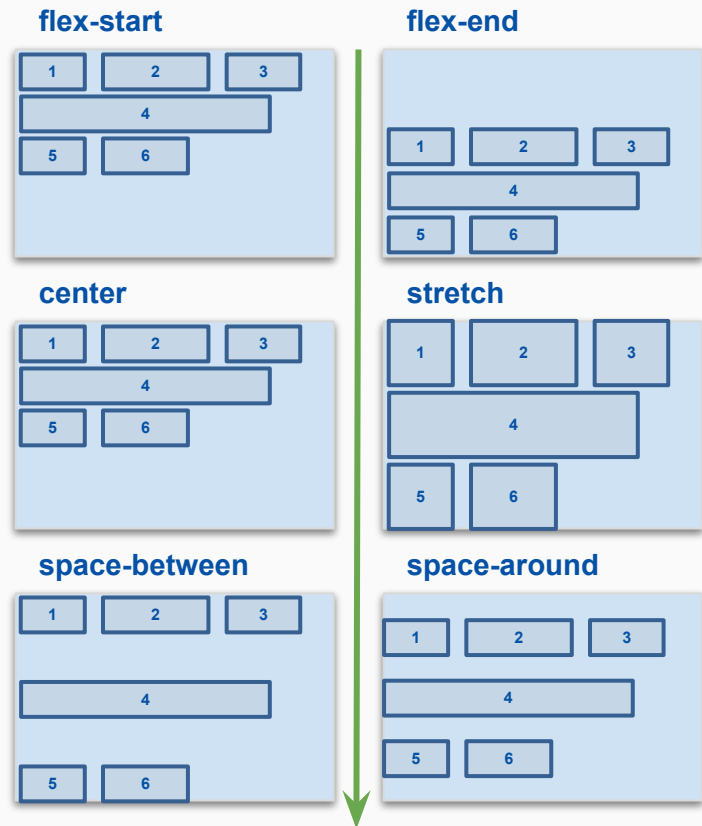
flex-wrap : nowrap



flex-wrap : wrap



flex-wrap : wrap-reverse

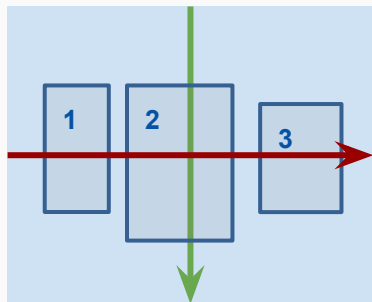


`align-content` - это свойство выравнивает **линии** в пределах flex контейнера, подобно тому, как `justify-content` выравнивает отдельные элементы в пределах главной оси.

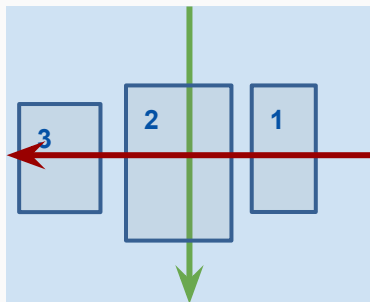
Не действует, когда есть только одна строка flex элементов

flex-direction

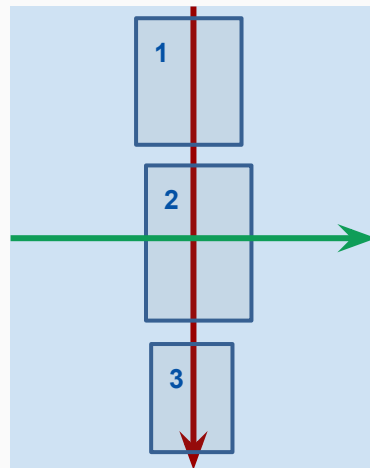
flex-direction - устанавливает основную ось, таким образом определяя направление flex элементов, помещаемых в flex контейнер



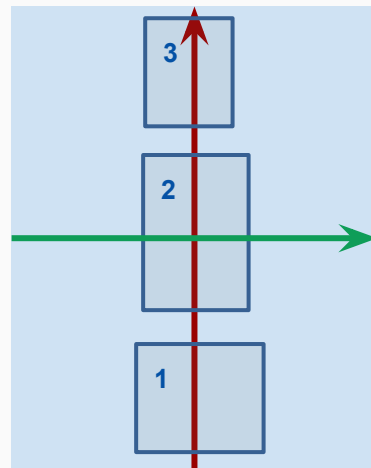
flex-direction : row



flex-direction : row-reverse



flex-direction : column



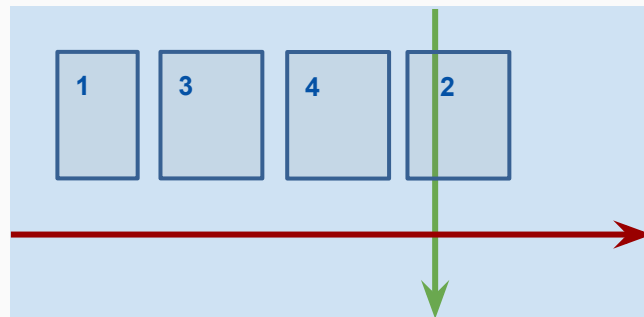
**flex-direction :
column-reverse**

Свойства flex элементов

order

order - определяет порядок вывода элементов внутри flex контейнер

- по-умолчанию, значение `order` равно 0
- элементы располагаются от меньшего к большему
- При равных значениях `order` элементы выводятся в том порядке, в каком они появляются в исходном коде.



HTML

```
<nav class="container">
  <div class="flex1">1</div>
  <div class="flex2">2</div>
  <div class="flex3">3</div>
  <div class="flex3">4</div>
</nav>
```

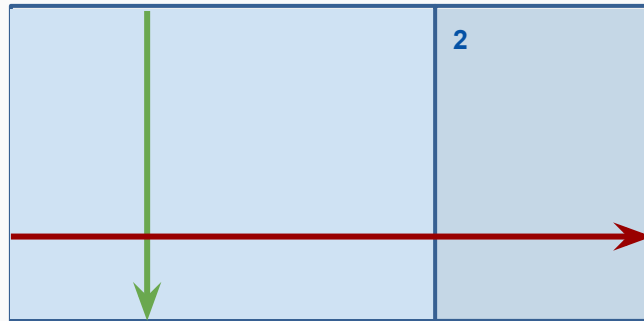
CSS

```
.container { display: flex; }
.flex1 { order: 5; }
.flex2 { order: 15; }
.flex3 { order: 5; }
.flex4 { order: 10; }
```

flex-grow

flex-grow - коэффициент, насколько элемент может вырасти относительно других flex-элементов в контейнере, **если есть свободное пространство**.

Если flex-grow равны, то всем элементам предоставляется одинаковое количество доступного пространства



HTML

```
<nav class="container">
  <div class="flex1">1</div>
  <div class="flex2">2</div>
</nav>
```

CSS

```
.container { display: flex; }
.flex1 { flex-grow: 3; }
.flex2 { flex-grow: 1; }
```

flex-basis, flex-grow, flex-shrink

flex-basis - начальный размер flex-элемента **вдоль главной оси**

flex-grow - коэффициент пространства контейнера выделяемое элементу. Если flex-grow равны, то всем элементам предоставляется одинаковое количество доступного пространства

flex-shrink - устанавливает коэффициент динамического сжатия флекс-элемента