

# CSS flexbox (Flexible Box Layout Module)

## модуль макета гибкого контейнера

- **способ компоновки элементов**

Flexbox состоит из:

**flex-контейнера** — родительского контейнера;

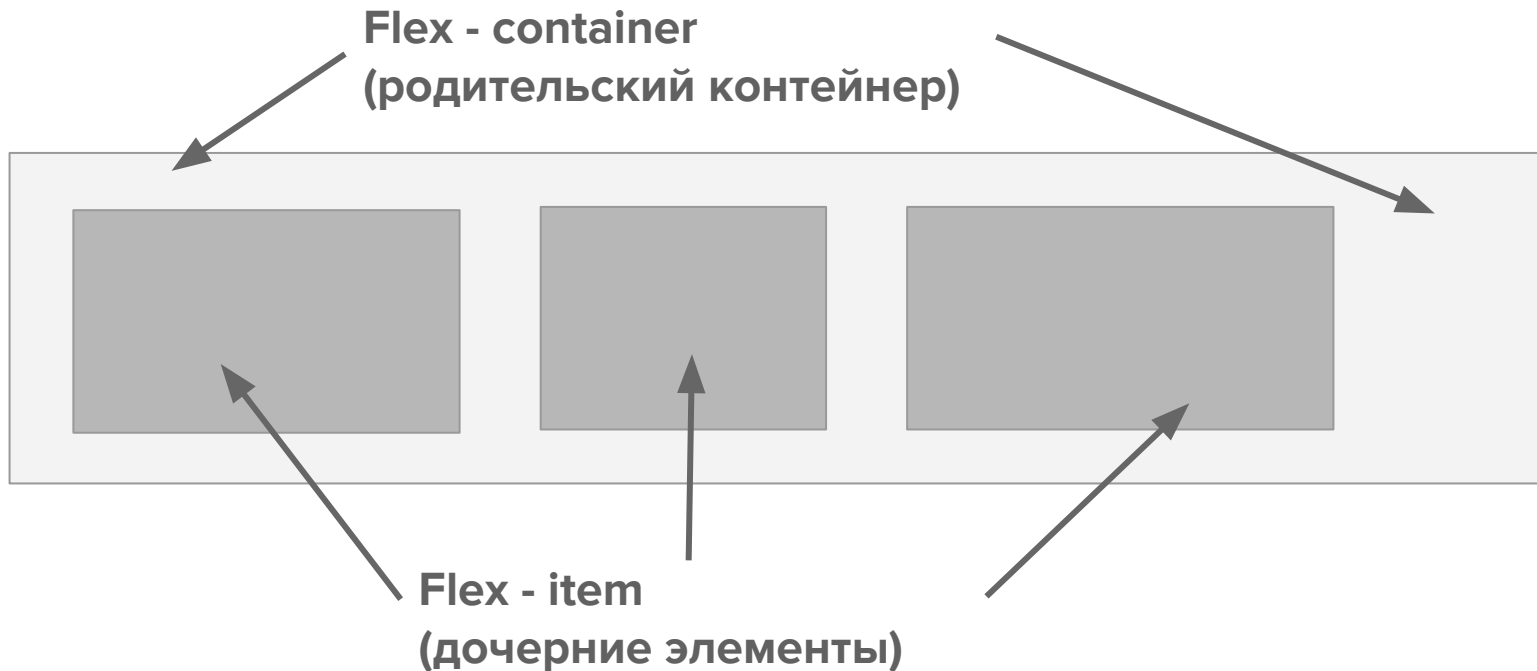
**flex-элементов** — дочерних блоков.

**Дочерние элементы** могут выстраиваться **в строку или столбик**, а оставшееся пространство распределяется между ними различными способами.

# Модуль flexbox позволяет:

- ❖ располагать элементы в одном из четырех направлений: слева направо, справа налево, сверху вниз или снизу вверх;
- ❖ переопределять порядок отображения элементов;
- ❖ автоматически определять размеры элементов таким образом, чтобы они вписывались в доступное пространство;
- ❖ решать проблему с горизонтальным и вертикальным центрированием;
- ❖ переносить элементы внутри контейнера, не допуская их переполнения;
- ❖ создавать колонки одинаковой высоты;
- ❖ создавать прижатый к низу footer сайта.

# Flexbox модель:



# Flexbox модель:

Для компоновки элементов с помощью модели flex-box нужно установить свойство **display** для родительского контейнера.

```
.flex-container {  
    display: flex; /* display: inline-flex; */  
}
```

После этого каждый дочерний элемент автоматически становится flex-элементом, выстраиваясь в ряд (вдоль главной оси) колонками одинаковой высоты, равной высоте блока-контейнера.

При этом ширина блоков равна ширине их содержимого с учетом внутренних полей и рамок элемента.

Текст или изображения без обертки, они становятся анонимными flex-элементами. Текст выравнивается по верхнему краю блока-контейнера, а высота изображения становится равной высоте блока.

# Управление дочерними элементами. Свойства родительского контейнера

1. **Выравнивание** элементов **по горизонтали justify-content** (по ширине flex-контейнера с распределением оставшегося пространства) [Слайд 8](#)

2. **Выравнивание** элементов **по вертикали align-items**

Свойство выравнивает flex-элементы, в том числе и анонимные по перпендикулярной оси (по высоте). [Слайд 9](#)

3. **Направление главной оси flex-direction** [Слайд 10](#)

Свойство определяет, каким образом flex-элементы укладываются во flex-контейнере, задавая направление главной оси flex-контейнера.

Они **могут располагаться: горизонтально или вертикально.**

Главная ось по умолчанию идет слева направо. Поперечная – сверху вниз.

# Управление дочерними элементами. Свойства родительского контейнера

## 4. Многострочность элементов flex-wrap Слайд 11

Отвечает за расположение дочерних элементов — в одну строку или в несколько, и направление, в котором будут укладываться новые строки.

По умолчанию flex-элементы укладываются в одну строку. При переполнении контейнера их содержимое будет выходить за границы flex-элементов.

**Краткая запись направления (flex-direction) и многострочности (flex-wrap) flex-flow:** направление многострочность;

# Управление дочерними элементами. Свойства родительского контейнера

## 5. Многострочное выравнивание align-content Слайд 12

Свойство выравнивает строки flex-элементов по вертикали во flex-контейнере, позволяя управлять свободным пространством.

**Свойство работает только в случае, если разрешен перенос строк, указано направление и высота flex-контейнера.**

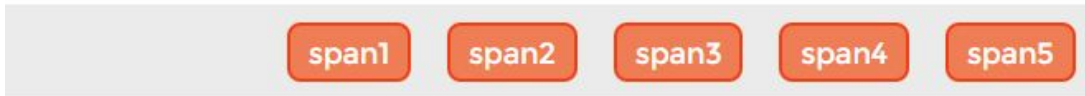
```
flex-flow: row/row-reverse/column/column-reverse wrap/wrap-reverse;  
height: 300px;
```

# Выравнивание по горизонтали

`justify-content: flex-start;`



`justify-content: flex-end;`



`justify-content: center;`



`justify-content: space-between;`



`justify-content: space-around;`



[Назад к презентации](#)



# Выравнивание по вертикали

`align-items: stretch;`



`align-items: flex-start;`



`align-items: flex-end;`



`align-items: center;`



`align-items: baseline;`



[Назад к презентации](#)

# Направление главной оси

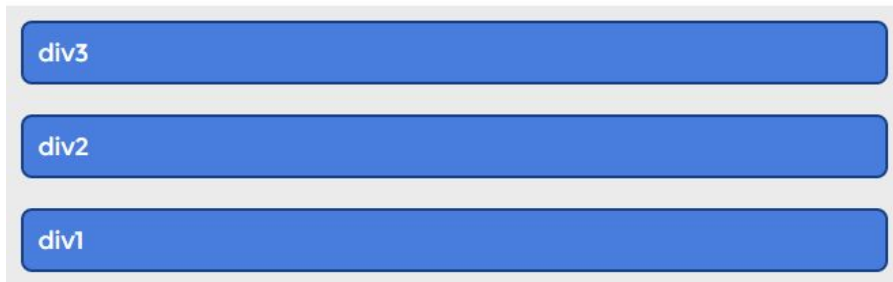
```
flex-direction: row-reverse;
```



```
flex-direction: column;
```



```
flex-direction: column-reverse;
```



[Назад к презентации](#)

# Многострочность элементов

`flex-wrap: wrap;`



`flex-wrap: wrap-reverse;`



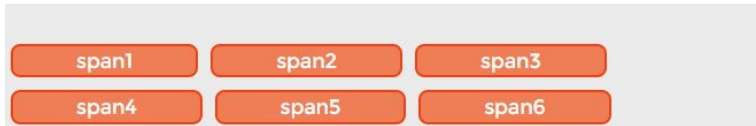
[Назад к презентации](#)

# Многострочное выравнивание

`align-content: flex-start;`



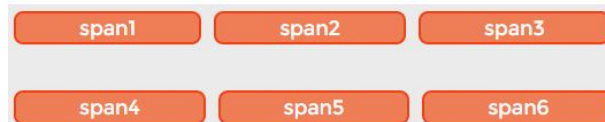
`align-content: flex-end;`



`align-content: center;`



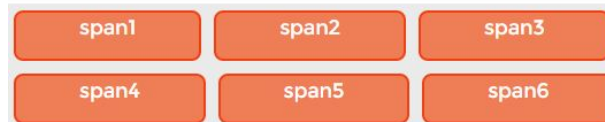
`align-content: space-between;`



`align-content: space-around;`



`align-content: stretch;`



[Назад к презентации](#)

# Управление дочерними элементами.

## Свойства дочерних элементов.

1. **Порядок отображения элементов order** - определяет порядок, в котором flex-элементы отображаются внутри flex-контейнера. [Слайд 17](#)

По умолчанию задан порядок `order: 0`; и они следуют друг за другом. Самый первый flex-элемент по умолчанию расположен слева.

Чтобы **переместить flex-элемент** ему нужно назначить `order`:

- ❖ в начало строки `order: -1`;
- ❖ в конец строки — `order: 1`.

2. **Базовая ширина элемента flex-basis** позволяет задать базовую ширину flex-элемента (px, %, em и тд), относительно которой будет происходить растяжение `flex-grow` или сужение `flex-shrink` элемента.

# Управление дочерними элементами.

## Свойства дочерних элементов.

3. **Растяжение элементов flex-grow** определяет коэффициент увеличения ширины flex-элемента относительно других flex-элементов. [Слайд 18](#)

Задается положительным целым или дробным числом, устанавливающим коэффициент увеличения flex-элемента (по умолчанию 0).

4. **Сужение элементов flex-shrink** указывает коэффициент уменьшения ширины flex-элемента относительно других flex-элементов. [Слайд 19](#)

**Работает только если для элемента задана ширина** с помощью свойства flex-basis или width.

Задается положительным целым или дробным числом, устанавливающим коэффициент уменьшения flex-элемента (по умолчанию 1).

# Управление дочерними элементами.

## Свойства дочерних элементов.

5. **Задание базовой ширины и трансформации элемента свойством `flex`** - сокращённая запись свойств `flex-grow`, `flex-shrink` и `flex-basis`.

Значение по умолчанию: `flex: 0 1 auto;`

Можно указывать как одно, так и все три значения свойств.

**Рекомендуется использовать сокращённую запись**, так как она правильно сбрасывает любые неуказанные компоненты.

# Управление дочерними элементами.

## Свойства дочерних элементов.

### Значения свойства:

- ❖ коэффициент растяжения - коэффициент увеличения ширины flex-элемента относительно других flex-элементов.
- ❖ коэффициент сужения - коэффициент уменьшения ширины flex-элемента относительно других flex-элементов.
- ❖ базовая ширина flex-элемента.
- ❖ auto = flex: 1 1 auto;.
- ❖ none = flex: 0 0 auto;.

6. **Выравнивание отдельно взятого flex-элемента по высоте** flex-контейнера - **align-self**. Переопределяет выравнивание, заданное align-items. **Слайд 20**



# Порядок отображения элементов



[Назад к презентации](#)

# Растяжение элементов



[Назад к презентации](#)

# Сужение элементов



[Назад к презентации](#)

# Выравнивание отдельных элементов

`align-self: flex-start;`

div2

div3

div4

div5

`align-self: flex-end;`

span2

span3

span4

span5

`align-self: center;`

div2

div3

div4

div5

`align-self: baseline;`

span2

span3

span4

span5

`align-self: stretch;`

div2

div3

div4

div5

[Назад к презентации](#)