

CSS flexbox

CSS flexbox (*Flexible Box Layout Module*) — модуль макета гибкого контейнера — представляет собой способ компоновки элементов, в основе лежит идея оси.

Flexbox состоит из **гибкого контейнера (flex container)** и **гибких элементов (flex items)**. Гибкие элементы могут выстраиваться в строку или столбик, а оставшееся свободное пространство распределяется между ними различными способами.

Модуль flexbox позволяет решать следующие задачи:

- 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 его переполнения.
- Создавать колонки одинаковой высоты.
- Создавать прижатый к низу страницы [подвал сайта](#).

Flexbox решает специфические задачи — создание одномерных макетов, например, навигационной панели, так как flex-элементы можно размещать только по одной из осей.

Список текущих проблем модуля и кросс-браузерных решений для них вы можете прочитать в статье Philip Walton [Flexbugs](#).

Поддержка браузерами

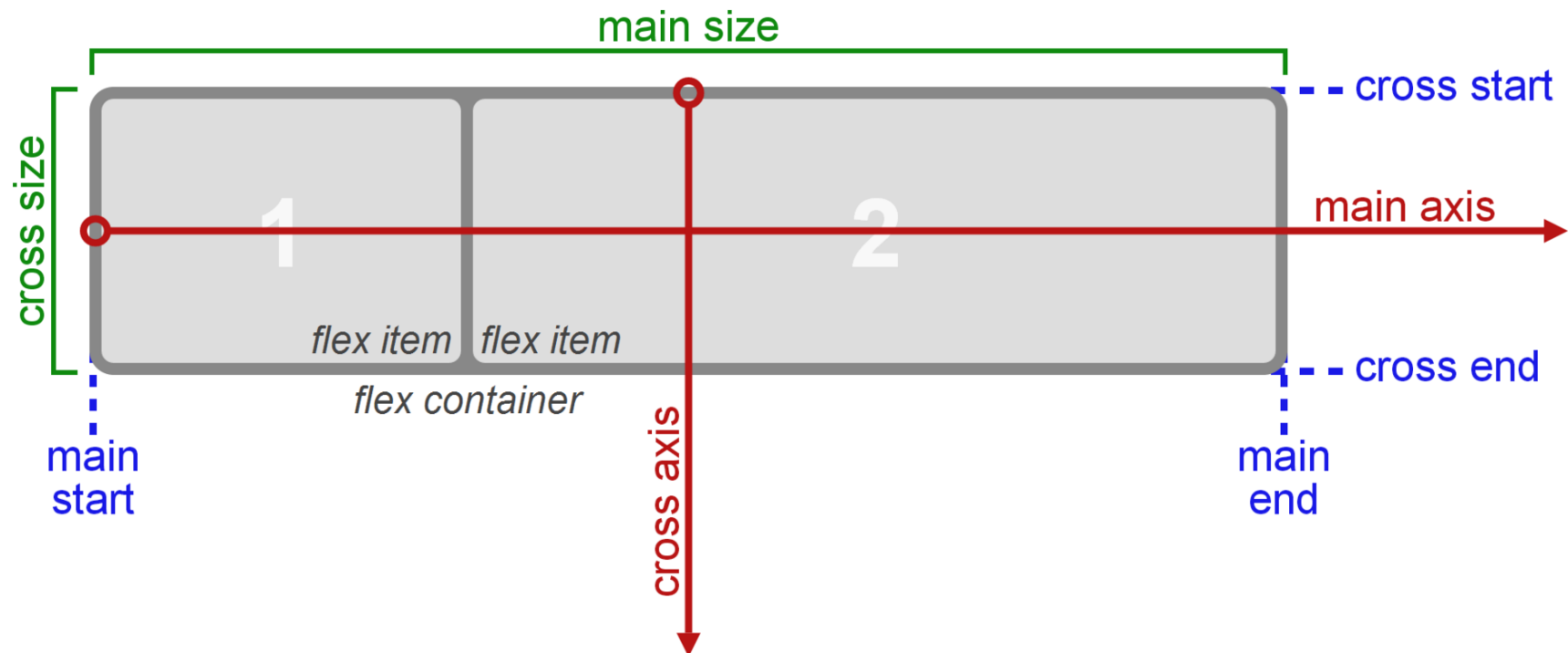


IE: 11.0, 10.0 -ms-

Firefox: 28.0, 18.0 -moz-

Chrome: 29.0, 21.0 -webkit-
Safari: 6.1 -webkit-
Opera: 12.1 -webkit-
iOS Safari: 7.0 -webkit-
Opera Mini: 8
Android Browser: 4.4, 4.1 -webkit-
Chrome for Android: 44

1. Основные понятия



📷 РИС. 1. МОДЕЛЬ FLEXBOX

Для описания модуля Flexbox используется определенный набор терминов. Значение `flex-flow` и режим записи определяют соответствие этих терминов физическим направлениям: верх / право / низ / лево, осям: вертикальная / горизонтальная и размерам: ширина / высота.

Главная ось (main axis) — ось, вдоль которой выкладываются flex-элементы. Она простирается в основном измерении.

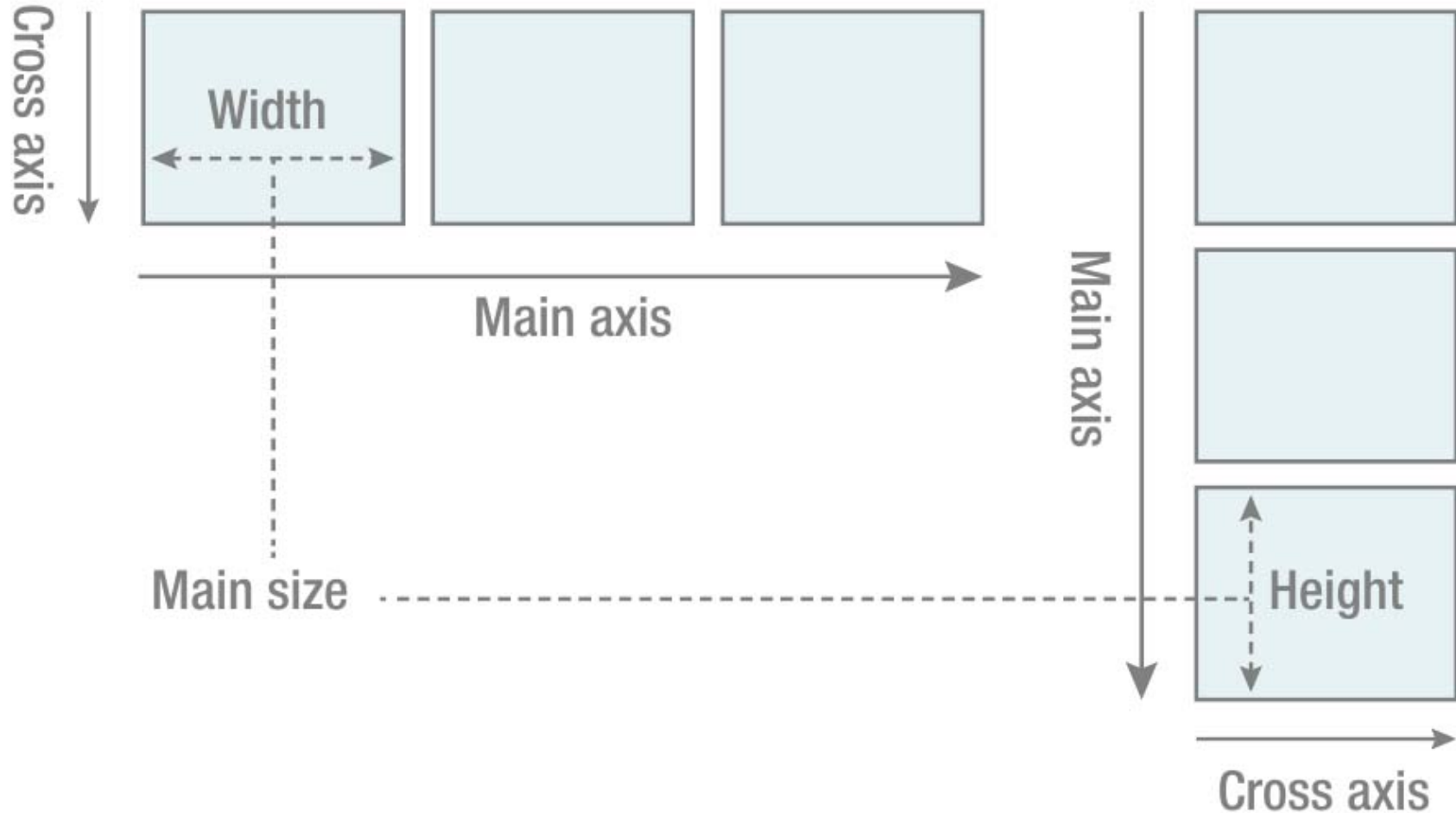
Main start и **main end** — линии, которые определяют начальную и конечную стороны flex-контейнера, относительно которых выкладываются flex-элементы (начиная с main start по направлению к main end).

Основной размер (main size) — ширина или высота flex-контейнера или flex-элементов, в зависимости от того, что из них находится в основном измерении, определяют основной размер flex-контейнера или flex-элемента.

Поперечная ось (cross axis) — ось, перпендикулярная главной оси. Она простирается в поперечном измерении.

Cross start и **cross end** — линии, которые определяют начальную и конечную стороны поперечной оси, относительно которых выкладываются flex-элементы.

Поперечный размер (cross size) — ширина или высота flex-контейнера или flex-элементов, в зависимости от того, что находится в поперечном измерении, являются их поперечным размером.



2. Flex-контейнер

Flex-контейнер устанавливает новый гибкий контекст форматирования для его содержимого. Flex-контейнер не является блочным контейнером, поэтому для дочерних элементов не работают такие CSS-свойства, как `float`, `clear`, `vertical-align`. Также, на flex-контейнер не оказывают влияние свойства `column-*`, создающие колонки в тексте и псевдоэлементы `::first-line` и `::first-letter`.

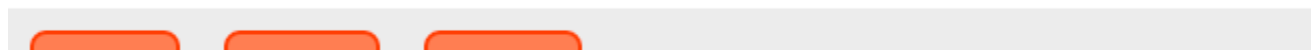
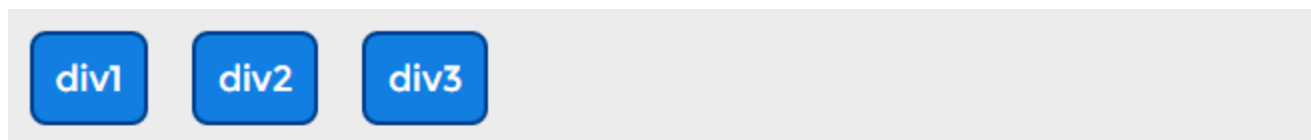
Модель flexbox-разметки связана с определенным значением CSS-свойства `display` родительского html-элемента, содержащего внутри себя дочерние блоки. Для управления элементами с помощью этой модели нужно установить свойство `display` следующим образом:

CSS

```
.flex-container {
/*генерирует flex-контейнер уровня блока*/
display: -webkit-flex;
display: flex;
}

.flex-container {
/*генерирует flex-контейнер уровня строки*/
display: -webkit-inline-flex;
display: inline-flex;
}
```

После установки данных значений свойства каждый дочерний элемент автоматически становится flex-элементом, выстраиваясь в один ряд (вдоль главной оси). При этом блочные и строчные дочерние элементы ведут себя одинаково, т.е. ширина блоков равна ширине их содержимого с учетом внутренних полей и рамок элемента.





📷 РИС. 3. ВЫРАВНИВАНИЕ ЭЛЕМЕНТОВ В МОДЕЛИ FLEXBOX

Если родительский блок содержит текст или изображения без оберток, они становятся анонимными flex-элементами. Текст выравнивается по верхнему краю блока-контейнера, а высота изображения становится равной высоте блока, т.е. оно деформируется.

3. Flex-элементы

Flex-элементы — блоки, представляющие содержимое flex-контейнера в потоке. Flex-контейнер устанавливает новый контекст форматирования для своего содержимого, который обуславливает следующие особенности:

- Для flex-элементов блокируется их значение свойства `display`. Значение `display: inline-block;` и `display: table-cell;` вычисляется в `display: block;`.
- Пустое пространство между элементами исчезает: оно не становится своим собственным flex-элементом, даже если межэлементный текст обернут в анонимный flex-элемент. Для содержимого анонимного flex-элемента невозможно задать собственные стили, но оно будет наследовать их (например, параметры шрифта) от flex-контейнера.
- Абсолютно позиционированный flex-элемент не участвует в компоновке гибкого макета.
- Поля `margin` соседних flex-элементов не схлопываются.
- Процентные значения `margin` и `padding` вычисляются от внутреннего размера содержащего их блока.
- `margin: auto;` расширяются, поглощая дополнительное пространство в соответствующем измерении. Их можно использовать для выравнивания или раздвигания смежных flex-элементов.
- Автоматический минимальный размер flex-элементов по умолчанию является минимальным размером его содержимого, то есть `min-width: auto;`. Для контейнеров с прокруткой автоматический минимальный размер обычно равен нулю.

4. Порядок отображения flex-элементов и ориентация

Содержимое flex-контейнера можно разложить в любом направлении и в любом порядке (переупорядочение flex-элементов внутри контейнера влияет только на визуальный рендеринг).

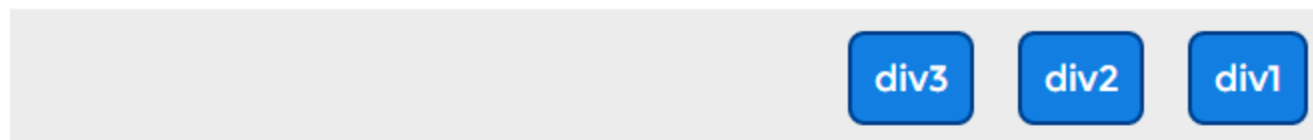
4.1. Направление главной оси: flex-direction

Свойство относится к flex-контейнеру. Управляет направлением главной оси, вдоль которой укладываются flex-элементы, в соответствии с текущим режимом записи.

Свойство не наследуется.

flex-direction	
row	Значение по умолчанию, слева направо (в rtl справа налево). Flex-элементы выкладываются в строку. Начало (main-start) и конец (main-end) направления главной оси соответствуют началу (inline-start) и концу (inline-end) оси строки (inline-axis).
row-reverse	Направление справа налево (в rtl слева направо). Flex-элементы выкладываются в строку относительно правого края контейнера (в rtl — левого).
column	Направление сверху вниз. Flex-элементы выкладываются в колонку.
column-reverse	Колонка с элементами в обратном порядке, снизу вверх.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

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



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





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



📷 РИС. 4. СВОЙСТВО FLEX-DIRECTION ДЛЯ LEFT-TO-RIGHT ЯЗЫКОВ

Синтаксис

```
flex-direction: row;  
flex-direction: row-reverse;  
flex-direction: column;  
flex-direction: column-reverse;  
flex-direction: inherit;  
flex-direction: initial;
```

CSS

4.2. Управление многострочностью flex-контейнера: flex-wrap

Свойство определяет, будет ли flex-контейнер однострочным или многострочным, а также задает направление

поперечной оси, определяющее направление укладки новых линий flex-контейнера.

По умолчанию flex-элементы укладываются в одну строку, вдоль главной оси. При переполнении они будут выходить за пределы ограничивающей рамки flex-контейнера.

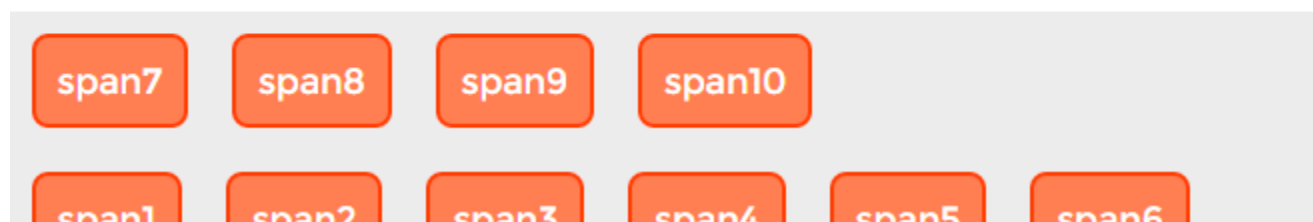
Свойство не наследуется.


flex-wrap	
nowrap	Значение по умолчанию. Flex-элементы не переносятся, а располагаются в одну линию слева направо (в rtl справа налево).
wrap	Flex-элементы переносятся, располагаясь в несколько горизонтальных рядов (если не помещаются в один ряд) в направлении слева направо (в rtl справа налево).
wrap-reverse	Flex-элементы переносятся на новые линии, располагаясь в обратном порядке слева-направо, при этом перенос происходит снизу вверх.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

`flex-wrap: wrap;`



`flex-wrap: wrap-reverse;`



 РИС. 5. УПРАВЛЕНИЕ МНОГОСТРОЧНОСТЬЮ С ПОМОЩЬЮ СВОЙСТВА FLEX-WRAP ДЛЯ LTR-ЯЗЫКОВ

Синтаксис

```
flex-wrap: nowrap;  
flex-wrap: wrap;  
flex-wrap: wrap-reverse;  
flex-wrap: inherit;  
flex-wrap: initial;
```

CSS

4.3. Краткая запись направления и многострочности: flex-flow

Свойство позволяет определить направления главной и поперечной осей, а также возможность переноса flex-элементов при необходимости на несколько строк. Представляет собой сокращённую запись свойств flex-direction и flex-wrap. Значение по умолчанию flex-flow: row nowrap; .

Свойство не наследуется.

flex-flow

направление	Указывает направление главной оси. Значение по умолчанию row .
многострочность	Задаёт многострочность поперечной оси. Значение по умолчанию nowrap .
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

Синтаксис

```
flex-flow: row;
```

CSS

```
flex-flow: row-reverse;
flex-flow: column;
flex-flow: column-reverse;
flex-flow: nowrap;
flex-flow: wrap;
flex-flow: wrap-reverse;
flex-flow: row nowrap;
flex-flow: column wrap;
flex-flow: column-reverse wrap-reverse;
flex-flow: inherit;
flex-flow: initial;
```

4.4. Порядок отображения flex-элементов: order

Свойство определяет порядок, в котором flex-элементы отображаются и располагаются внутри flex-контейнера. Применяется к flex-элементам.

Свойство не наследуется.

Первоначально все flex-элементы имеют `order: 0`; . При указании значения от -1 для элемента он перемещается в начало строки, значение 1 — в конец. Если несколько flex-элементов имеют одинаковое значение `order` , они будут отображаться в соответствии с исходным порядком.

order	
число	Свойство задается целым числом, отвечающим за порядок отображения flex-элементов. Значение по умолчанию 0 .
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

Синтаксис

```

order: 2;
order: -1;
order: inherit;
order: initial;

```



📷 РИС. 6. ПОРЯДОК ОТОБРАЖЕНИЯ FLEX-ЭЛЕМЕНТОВ

5. Гибкость flex-элементов

Определяющим аспектом гибкого макета является возможность «сгибать» flex-элементы, изменяя их ширину / высоту (в зависимости от того, какой размер находится на главной оси), чтобы заполнить доступное пространство в основном измерении. Это делается с помощью свойства `flex`. Flex-контейнер распределяет свободное пространство между своими дочерними элементами (пропорционально их коэффициенту `flex-grow`) для заполнения контейнера или сжимает их (пропорционально их коэффициенту `flex-shrink`), чтобы предотвратить переполнение.

Flex-элемент будет полностью «негибок», если его значения `flex-grow` и `flex-shrink` равны нулю, и «гибкий» в противном случае.

5.1. Задание гибких размеров одним свойством: `flex`

Свойство является сокращённой записью 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.

Свойство не наследуется.

W3C рекомендует использовать сокращённую запись, так как она правильно сбрасывает любые неуказанные компоненты, чтобы подстроиться под типичное использование.

flex

коэффициент растяжения	Коэффициент увеличения ширины flex-элемента относительно других flex-элементов.
коэффициент сужения	Коэффициент уменьшения ширины flex-элемента относительно других flex-элементов.
базовая ширина	Базовая ширина flex-элемента.
auto	Эквивалентно <code>flex: 1 1 auto; .</code>
none	Эквивалентно <code>flex: 0 0 auto; .</code>
initial	Устанавливает значение свойства в значение по умолчанию. Эквивалентно <code>flex: 0 1 auto; .</code>
inherit	Наследует значение свойства от родительского элемента.

Синтаксис

CSS

```
flex: none;
/* Одно значение, число без единиц: flex-grow */
flex: 2;
/* Одно значение, ширина/высота: flex-basis */
flex: 10em;
flex: 30px;
flex: auto;
flex: content;
/* Два значения: flex-grow | flex-basis */
flex: 1 30px;
/* Два значения: flex-grow | flex-shrink */
flex: 2 2;
/* Три значения: flex-grow | flex-shrink | flex-basis */
```

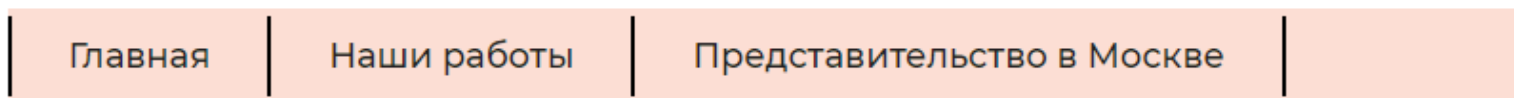
```
flex: 2 2 10%;
flex: inherit;
flex: initial;
```

5.2. Коэффициент роста: flex-grow

Свойство определяет коэффициент роста одного flex-элемента относительно других flex-элементов в flex-контейнере при распределении положительного свободного пространства. Если сумма значений flex-grow flex-элементов в строке меньше 1, они занимают менее 100% свободного пространства.

Свойство не наследуется.

flex-grow	
число	Положительное целое или дробное число, устанавливающее коэффициент роста flex-элемента. Значение по умолчанию 0.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.



```
li {flex-grow: 0;}
```



```
li {flex-grow: 1;}
```



```
li:last-child {flex-grow: 1;}
```

📷 РИС. 7. УПРАВЛЕНИЕ СВОБОДНЫМ ПРОСТРАНСТВОМ В ПАНЕЛИ НАВИГАЦИИ С ПОМОЩЬЮ FLEX-GROW

Синтаксис

CSS

```
flex-grow: 3;  
flex-grow: 0.6;  
flex-grow: inherit;  
flex-grow: initial;
```

5.3. Коэффициент сжатия: flex-shrink

Свойство указывает коэффициент сжатия flex-элемента относительно других flex-элементов при распределении отрицательного свободного пространства. Умножается на базовый размер `flex-basis`. Отрицательное пространство распределяется пропорционально тому, насколько элемент может сжаться, поэтому, например, маленький flex-элемент не уменьшится до нуля, пока не будет заметно уменьшен flex-элемент большего размера.

Свойство не наследуется.

flex-shrink

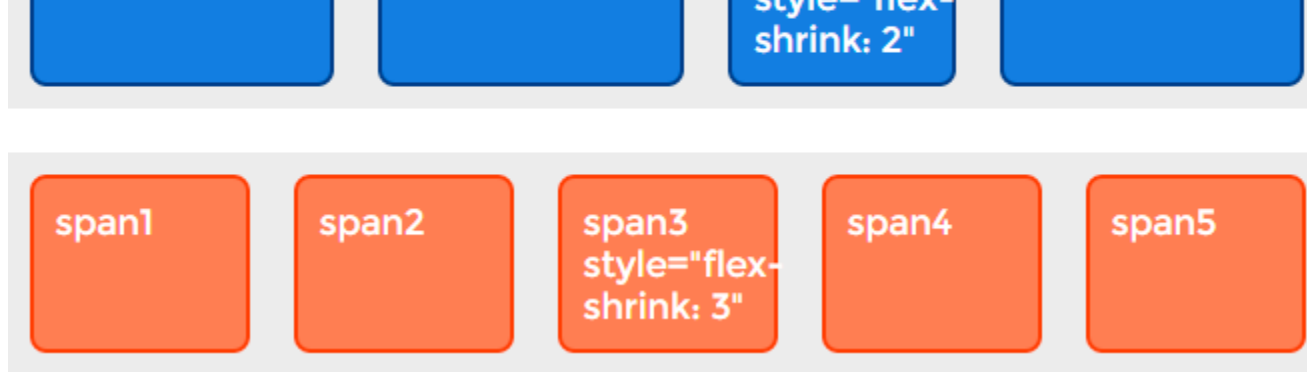
число	Положительное целое или дробное число, устанавливающее коэффициент уменьшения flex-элемента. Значение по умолчанию 1 .
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

div1

div2

div3
style="flex-

div4



📷 РИС. 8. СУЖЕНИЕ FLEX-ЭЛЕМЕНТОВ В СТРОКЕ

Синтаксис

CSS

```
flex-shrink: 2;  
flex-shrink: 0.6;  
flex-shrink: inherit;  
flex-shrink: initial;
```

5.4. Базовый размер: flex-basis

Свойство устанавливает начальный основной размер flex-элемента до распределения свободного пространства в соответствии с коэффициентами гибкости. Для всех значений, кроме `auto` и `content`, базовый гибкий размер определяется так же, как `width` в горизонтальных режимах записи. Процентные значения определяются относительно размера flex-контейнера, а если размер не задан, используемым значением для `flex-basis` являются размеры его содержимого.

Свойство не наследуется.

flex-basis	
auto	Значение по умолчанию. Элемент получает базовый размер, соответствующий размеру его содержимого (если он не задан явно).
content	Определяет базовый размер в зависимости от содержимого flex-элемента.

длина	Базовый размер определяется так же, как для ширины и высоты. Задается в единицах длины.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

Синтаксис

CSS

```
flex-basis: 10em;
flex-basis: 3px;
flex-basis: auto;
flex-basis: content;
flex-basis: inherit;
flex-basis: initial;
```

6. Выравнивание

6.1. Выравнивание по главной оси: justify-content

Свойство выравнивает flex-элементы по главной оси flex-контейнера, распределяя свободное пространство, незанятое flex-элементами. Когда элемент преобразуется в flex-контейнер, flex-элементы по умолчанию сгруппированы вместе (если для них не заданы поля `margin`). Промежутки добавляются после расчета значений `margin` и `flex-grow`. Если какие-либо элементы имеют ненулевое значение `flex-grow` или `margin: auto;`, свойство не будет оказывать влияния.

Свойство не наследуется.

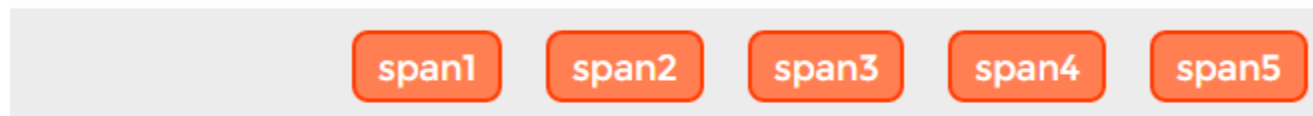
justify-content	
flex-start	Значение по умолчанию. Flex-элементы выкладываются в направлении, идущем от начала flex-контейнера.
flex-end	Flex-элементы размещаются в конце flex-контейнера.

center	Flex-элементы выравниваются по центру flex-контейнера.
space-between	Flex-элементы равномерно распределяются по линии. Первый flex-элемент помещается вровень с краем начальной линии, последний flex-элемент — вровень с краем конечной линии, а остальные flex-элементы на линии распределяются так, чтобы расстояние между любыми двумя соседними элементами было одинаковым. Если оставшееся свободное пространство отрицательно или в строке присутствует только один flex-элемент, это значение идентично параметру <code>flex-start</code> .
space-around	Flex-элементы на линии распределяются так, чтобы расстояние между любыми двумя смежными flex-элементами было одинаковым, а расстояние между первым / последним flex-элементами и краями flex-контейнера составляло половину от расстояния между flex-элементами.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

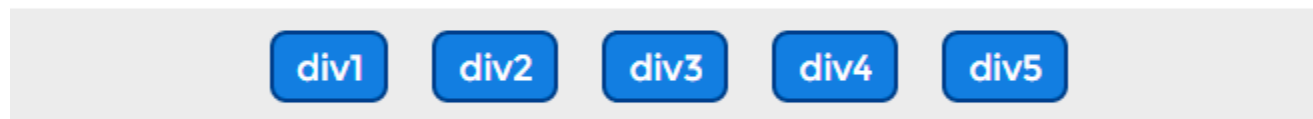
`justify-content: flex-start;`



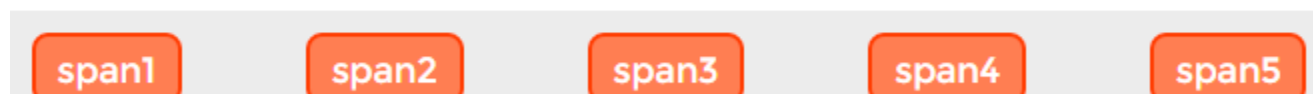
`justify-content: flex-end;`



`justify-content: center;`



`justify-content: space-between;`



```
justify-content: space-around;
```



📷 РИС. 9. ВЫРАВНИВАНИЕ ЭЛЕМЕНТОВ И РАСПРЕДЕЛЕНИЕ СВОБОДНОГО ПРОСТРАНСТВА С ПОМОЩЬЮ СВОЙСТВА JUSTIFY-CONTENT

Синтаксис

CSS

```
justify-content: center;  
justify-content: flex-start;  
justify-content: flex-end;  
justify-content: space-between;  
justify-content: space-around;  
justify-content: stretch;  
justify-content: inherit;  
justify-content: initial;
```

6.2. Выравнивание по поперечной оси: align-items и align-self

Flex-элементы можно выравнивать по поперечной оси текущей строки flex-контейнера. `align-items` устанавливает выравнивание для всех элементов flex-контейнера, включая анонимные flex-элементы. `align-self` позволяет переопределить это выравнивание для отдельных flex-элементов. Если любое из поперечных margin flex-элемента имеет значение `auto`, `align-self` не имеет никакого влияния.

6.2.1. align-items

Свойство выравнивает flex-элементы, в том числе и анонимные flex-элементы по поперечной оси.

Свойство не наследуется.

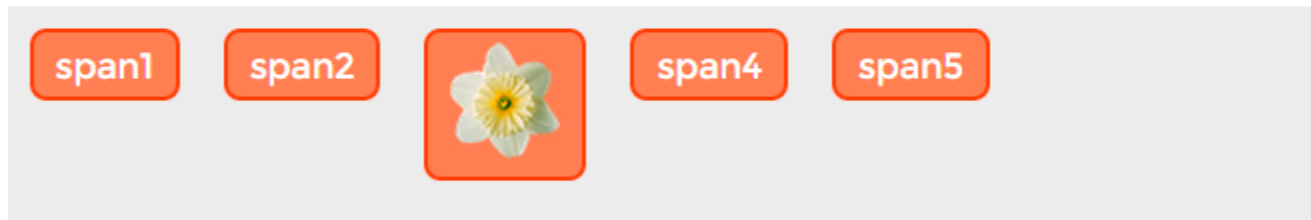
align-items

flex-start	Верхний край flex-элемента помещается вплотную с flex-линией (или на расстоянии, с учетом заданных полей <code>margin</code> и рамок <code>border</code> элемента), проходящей через начало поперечной оси.
flex-end	Нижний край flex-элемента помещается вплотную с flex-линией (или на расстоянии, с учетом заданных полей <code>margin</code> и рамок <code>border</code> элемента), проходящей через конец поперечной оси.
center	Поля flex-элемента центрируются по поперечной оси в пределах flex-линии.
baseline	Базовые линии всех flex-элементов, участвующих в выравнивании, совпадают.
stretch	Если поперечный размер flex-элемента вычисляется как <code>auto</code> и ни одно из поперечных значений <code>margin</code> не равно <code>auto</code> , элемент растягивается. Значение по умолчанию.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

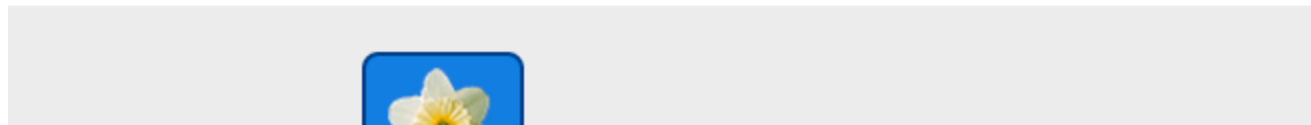
`align-items: stretch;`



`align-items: flex-start;`

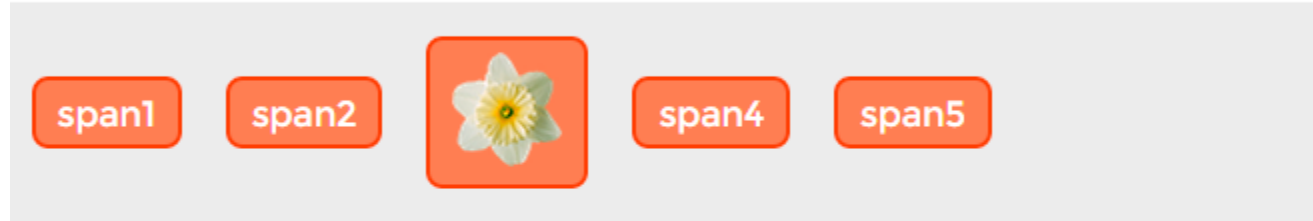


`align-items: flex-end;`





```
align-items: center;
```



```
align-items: baseline;
```



📷 РИС. 10. ВЫРАВНИВАНИЕ ЭЛЕМЕНТОВ В КОНТЕЙНЕРЕ ПО ВЕРТИКАЛИ

Синтаксис

CSS

```
align-items: stretch;  
align-items: center;  
align-items: flex-start;  
align-items: flex-end;  
align-items: baseline;  
align-items: inherit;  
align-items: initial;
```

6.2.2. align-self

Свойство отвечает за выравнивание отдельно взятого flex-элемента по высоте flex-контейнера. Переопределяет выравнивание, заданное `align-items`.

Свойство не наследуется.

align-self

auto	Значение по умолчанию. Flex-элемент использует выравнивание, указанное в свойстве <code>align-items</code> flex-контейнера.
flex-start	Верхний край flex-элемента помещается вплотную с flex-линией (или на расстоянии, с учетом заданных полей <code>margin</code> и рамок <code>border</code> элемента), проходящей через начало поперечной оси.
flex-end	Нижний край flex-элемента помещается вплотную с flex-линией (или на расстоянии, с учетом заданных полей <code>margin</code> и рамок <code>border</code> элемента), проходящей через конец поперечной оси.
center	Поля flex-элемента центрируются по поперечной оси в пределах flex-линии.
baseline	Flex-элемент выравнивается по базовой линии.
stretch	Если поперечный размер flex-элемента вычисляется как <code>auto</code> и ни одно из поперечных значений <code>margin</code> не равно <code>auto</code> , элемент растягивается. Значение по умолчанию.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.

align-self: flex-start;

div2

div3

div4

div5

align-self: flex-end;

span2

span3

span4

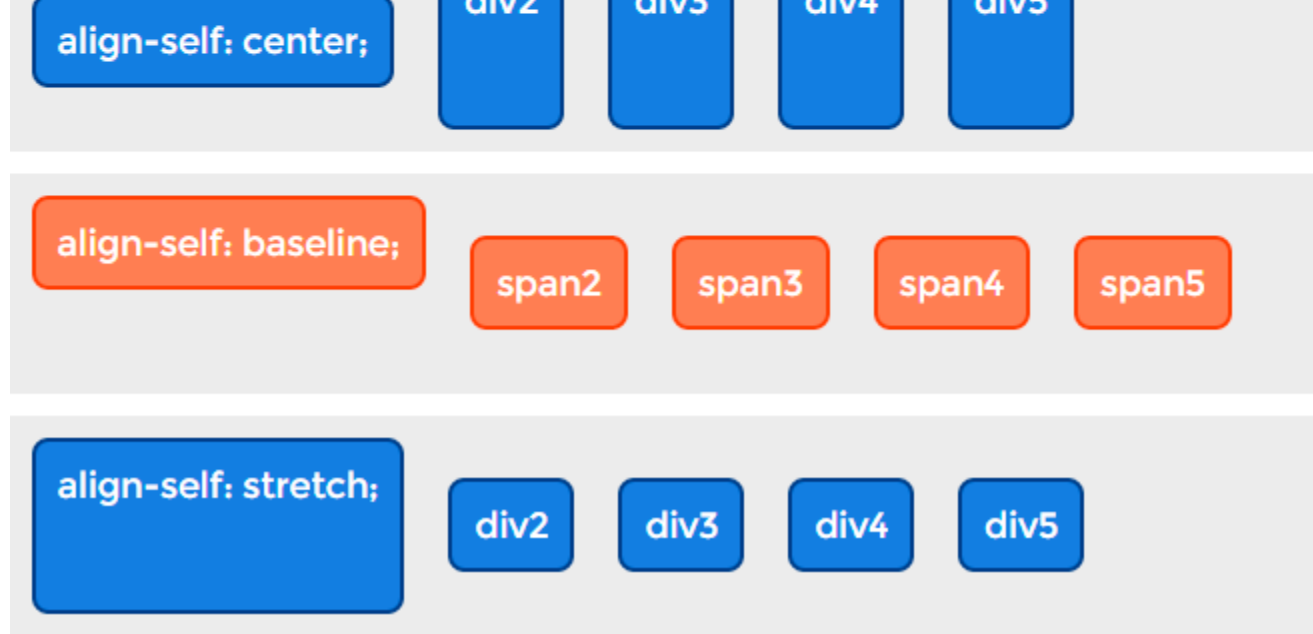
span5

div2

div3

div4

div5



📷 РИС. 11. ВЫРАВНИВАНИЕ ОТДЕЛЬНЫХ FLEX-ЭЛЕМЕНТОВ

Синтаксис

```
align-self: auto;  
align-self: center;  
align-self: flex-start;  
align-self: flex-end;  
align-self: baseline;  
align-self: stretch;  
align-self: inherit;  
align-self: initial;
```

CSS

6.3. Выравнивание строк flex-контейнера: align-content

Свойство выравнивает строки в flex-контейнере при наличии дополнительного пространства на поперечной оси, аналогично выравниванию отдельных элементов на главной оси с помощью свойства `justify-content`. Свойство не влияет на однострочный flex-контейнер.

Свойство не наследуется.

align-content

flex-start	Строки укладываются по направлению к началу flex-контейнера. Край первой строки помещается вплотную к краю flex-контейнера, каждая последующая — вплотную к предыдущей строке.
flex-end	Строки укладываются по направлению к концу flex-контейнера. Край последней строки помещается вплотную к краю flex-контейнера, каждая предыдущая — вплотную к последующей строке.
center	Строки укладываются по направлению к центру flex-контейнера. Строки расположены вплотную друг к другу и выровнены по центру flex-контейнера с равным расстоянием между начальным краем содержимого flex-контейнера и первой строкой и между конечным краем содержимого flex-контейнера и последней строкой.
space-between	Строки равномерно распределены в flex-контейнере. Если оставшееся свободное пространство отрицательно или в flex-контейнере имеется только одна flex-линия, это значение идентично flex-start . В противном случае край первой строки помещается вплотную к начальному краю содержимого flex-контейнера, край последней строки — вплотную к конечному краю содержимого flex-контейнера. Остальные строки распределены так, чтобы расстояние между любыми двумя соседними строками было одинаковым.
space-around	Строки равномерно распределены в flex-контейнере с половинным пробелом на обоих концах. Если оставшееся свободное пространство отрицательно, это значение идентично center . В противном случае строки распределяются таким образом, чтобы расстояние между любыми двумя соседними строками было одинаковым, а расстояние между первой / последней строками и краями содержимого flex-контейнера составляло половину от расстояния между строками.
stretch	Значение по умолчанию. Строки flex-элементов равномерно растягиваются, заполняя все доступное пространство. Если оставшееся свободное пространство отрицательно, это значение идентично flex-start . В противном случае свободное пространство будет разделено поровну между всеми строками, увеличивая их поперечный размер.
initial	Устанавливает значение свойства в значение по умолчанию.

inherit

Наследует значение свойства от родительского элемента.

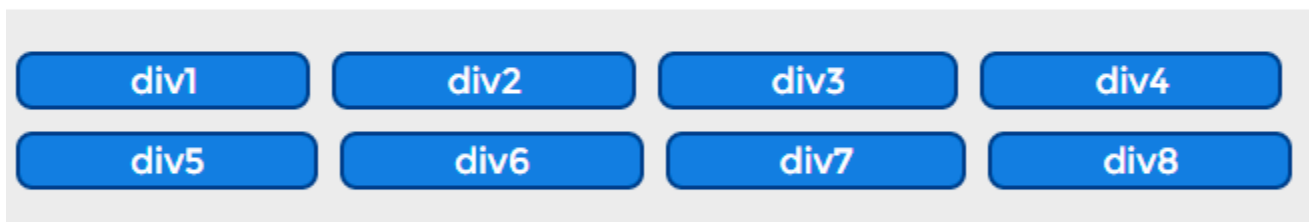
`align-content: flex-start;`



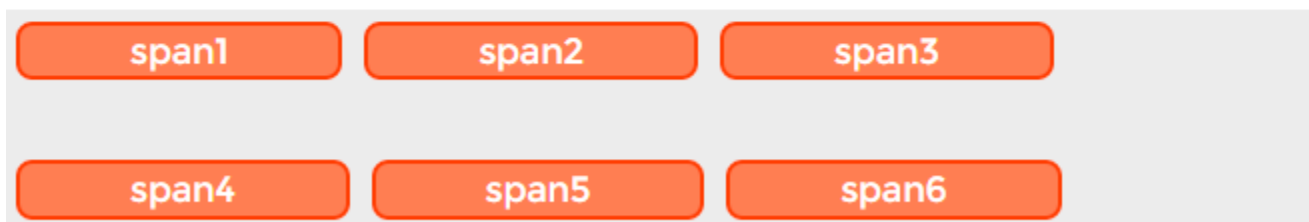
`align-content: flex-end;`



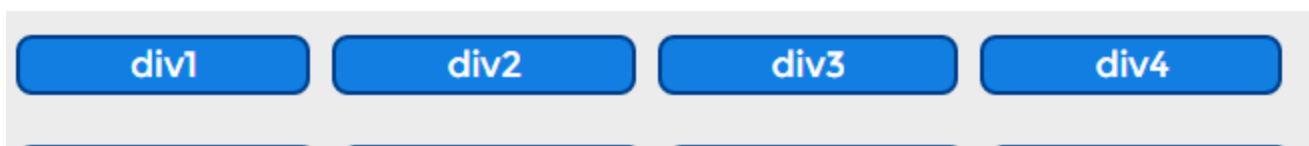
`align-content: center;`

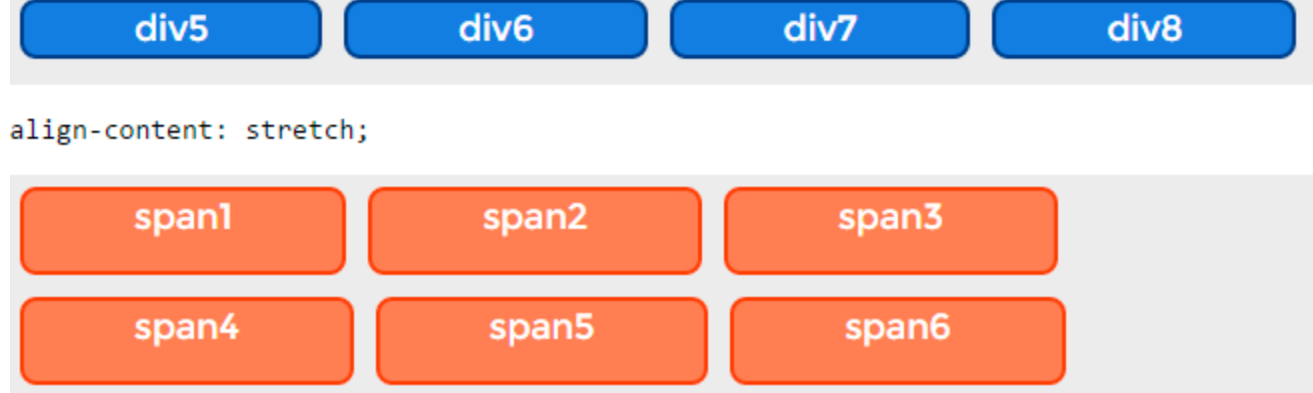


`align-content: space-between;`



`align-content: space-around;`





📷 РИС. 12. МНОГОСТРОЧНОЕ ВЫРАВНИВАНИЕ FLEX-ЭЛЕМЕНТОВ

Синтаксис

CSS

```
align-content: center;  
align-content: flex-start;  
align-content: flex-end;  
align-content: space-between;  
align-content: space-around;  
align-content: stretch;  
align-content: inherit;  
align-content: initial;
```