**Флексбокс**

Флексбокс — это CSS-механизм, который позволяет контролировать размер, порядок и выравнивание элементов по нескольким осям, распределение свободного места между элементами и многое другое.

Чтобы включить флексбокс, нужно задать элементу свойство display: flex;. После этого:

1. Элемент с display: flex; превращается во *«флекс-контейнер»*.
2. Непосредственные потомки этого элемента превращаются во *«флекс-элементы»* и начинают распределяться по новым правилам.

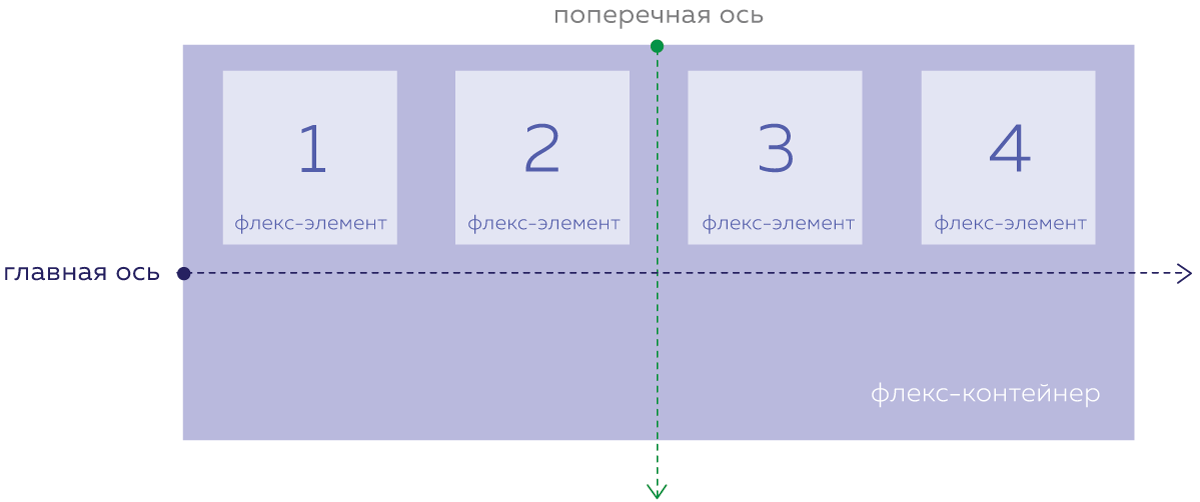
**Главная и поперечная оси**

Оси — одно из основных понятий во флексбоксах.

В обычном потоке документа блоки и текст располагаются слева направо и сверху вниз.

В привычной блочной модели направления «лево», «право», «верх» и «низ» неизменны. Но внутри флекс-контейнера эти понятия могут изменяться, потому что там можно изменять обычное направление потока.

* **Главной осью** flex-контейнера является направление, в соответствии с которым располагаются все его дочерние элементы. Поток флекс-элементов «течёт» вдоль главной оси от её начала к её концу.
* **Поперечной осью** называется направление, перпендикулярное главной оси. Вдоль этой оси работают «вертикальные» выравнивания.



По умолчанию главная ось направлена слева направо, но её можно разворачивать во всех направлениях с помощью свойства flex-direction, которое задаётся для флекс-контейнера. Значения свойства:

* Значение по умолчанию row — главная ось направлена слева направо.
* column — главная ось направлена сверху вниз.
* row-reverse — главная ось направлена справа налево.
* column-reverse — главная ось направлена снизу вверх.

Поперечная ось всегда перпендикулярна главной оси и поворачивается вместе с ней:

* Если главная ось направлена горизонтально, то поперечная ось смотрит вниз.
* Если главная ось направлена вертикально, то поперечная ось смотрит направо.

Таким образом, поперечная ось никогда не смотрит вверх или влево, и свойства для поворота поперечной оси нет.

**Распределение флекс-элементов**

*Выравнивание по главной оси*

CSS-свойство justify-content определяет то, как будут выровнены элементы вдоль **главной** оси. Доступные значения **justify-content**:

* Значение по умолчанию flex-start — элементы располагаются у начала главной оси.
* flex-end — элементы располагаются в конце главной оси.
* center — элементы располагаются по центру главной оси.
* space-between — элементы располагаются так, что расстояния между соседними одинаковые, а между элементами и краями флекс-контейнера отступов нет.
* space-around — элементы располагаются так, что расстояния между соседними одинаковые, а между элементами и краями флекс-контейнера есть отступ, равный половине расстояния между соседними элементами.

*Выравнивание по поперечной оси*

CSS-свойство align-items определяет то, как будут выровнены элементы вдоль поперечной оси. Доступные значения align-items:

* Значение по умолчанию stretch — элементы растягиваются на всю «высоту» флекс-контейнера.
* flex-start — элементы располагаются у начала поперечной оси.
* flex-end — элементы располагаются в конце поперечной оси.
* center — элементы располагаются по центру поперечной оси.
* baseline — элементы выравниваются по базовой линии текста внутри них.

Распределение элементов по главной оси задаётся для всего флекс-контейнера и на все флекс-элементы действует одинаково, задать какому-то элементу отличное от других распределение по главной оси нельзя.

Поперечное выравнивание можно задать каждому элементу отдельно. Для этого используется свойство align-self, которое задаётся для самих флекс-элементов, а не для флекс-контейнера. У свойства align-self те же самые значения, что и у align-items.

**Перенос флекс-элементов**

Если флекс-элементов в контейнере станет больше, чем может уместиться в один ряд, то:

* Они будут сжиматься до минимально возможной ширины.
* Даже если им задать ширину, механизм флексбокса может её уменьшить.
* Если они перестанут помещаться в контейнер и после уменьшения, то выйдут за его пределы, но продолжат располагаться в один ряд.

Чтобы этого не происходило, нужно воспользоваться свойством флекс-контейнера flex-wrap. У него есть два значения:

* Значение по умолчанию nowrap — перенос флекс-элементов на новую строку запрещён.
* wrap — разрешает перенос флекс-элементов на новую строку. Ряды элементов располагаются вдоль поперечной оси, первый ряд — в начале поперечной оси, последний — в конце.
* wrap-reverse — также разрешает перенос флекс-элементов на новую строку. Ряды элементов располагаются в обратном порядке: первый — в конце поперечной оси, последний — в начале.

**Выравнивание строк флекс-контейнера**

Свойство align-content управляет выравниванием рядов флекс-элементов вдоль поперечной оси. У него и свойства justify-content очень похожие значения:

* Значение по умолчанию stretch — растягивает ряды флекс-элементов, при этом оставшееся свободное место между ними делится поровну. Отображение строк при этом зависит от значения align-items:
  1. Если у align-items задано значение stretch, то элементы в строках растягиваются на всю высоту своей строки.
  2. Если значение отлично от stretch, то элементы в строках ужимаются под своё содержимое и выравниваются в строках в зависимости от значения align-items.
* flex-start — располагает ряды флекс-элементов в начале поперечной оси.
* flex-end — располагает ряды флекс-элементов в конце поперечной оси.
* center — располагает ряды флекс-элементов в середине поперечной оси так, что отступов между соседними рядами нет, а расстояния между первым рядом и краем флекс-контейнера равно расстоянию между последним рядом и другим краем.
* space-between — равномерно распределяет ряды флекс-элементов вдоль поперечной оси, расстояния между соседними рядами одинаковые, отступов у краёв нет.
* space-around — равномерно распределяет ряды флекс-элементов вдоль поперечной оси, расстояния между соседними рядами одинаковые, отступы у краёв равны половине расстояния между соседними рядами.

Свойство align-content «перекрывает» заданное значение align-items, которое управляет выравниванием флекс-элементов вдоль поперечной оси. Это происходит и в случае, когда есть только один ряд флекс-элементов, и когда рядов несколько.

Ранее в спецификации было описано другое поведение:

* Если есть только один ряд флекс-элементов, то работает align-items.
* Если есть несколько рядов, то работает align-content.

В начале 2019 года поведение было актуализировано согласно спецификации во всех современных браузерах, теперь его можно встретить только в старых браузерах.

**Порядковый номер флекс-элемента**

Порядок следования флекс-элементов в потоке можно изменять с помощью свойства order, порядкового номера флекс-элемента, не меняя при этом HTML-код.

По умолчанию порядковый номер флекс-элементов равен 0, а сортировка элементов производится по возрастанию номера. Порядковый номер задаётся целым числом, положительным или отрицательным.

**Применение флексбоксов**

Идеальное выравнивание

С помощью флексбокса можно отцентровать элемент по вертикали и горизонтали так, чтобы центровка сохранялась при изменении размеров элемента или контейнера.

Для этого нужно задать контейнеру раскладку флексбокса, а дочернему флекс-элементу margin: auto. В этом случае флекс-элемент уменьшит свой размер под содержимое и отцентруется по вертикали и горизонтали.

«Гибкое» меню

Флексбокс будет полезен, если нужно создать раскладку, в которой пункты равномерно распределены по блоку меню, при чём первый пункт примыкает к левой части блока меню, а последний — к правой, с небольшими внутренними отступами.

Чтобы это сделать, нужно задать меню раскладку флексбокса, тогда пункты станут флекс-элементами. Затем с помощью свойства распределения элементов justify-content: space-around; можно добиться нужного результата.

Если вы добавите в меню ещё один пункт, отступы между пунктами меню будут «гибко» меняться, подстраиваясь под новые условия.

Сортировка элементов на CSS

Используя одновременно флексбокс и селектор по выделению чекбокса :checked ~, можно с помощью этого селектора управлять порядком флекс-элементов, изменяя направление главной оси с помощью flex-direction.

Лучше всего эффект работает, когда направление главной оси меняется с «сверху вниз» на «снизу вверх». При этом флекс-контейнер должен находиться в разметке на одном уровне с чекбоксом.

Блоки одинаковой высоты

В обычной блочной модели есть фундаментальный недостаток — соседние блоки ничего не знают друг о друге, поэтому их высоты нельзя «связать». При этом надо учитывать, что содержимое блоков может быть разным и их высота может меняться.

На флексбоксах можно реализовать раскладку с блоками одинаковой высоты — флекс-элементы по умолчанию растягиваются на всю высоту контейнера. Для этого достаточно задать родительскому блоку display: flex;.