

Курс: HTML/CSS + JS С НУЛЯ

Тренер:
Илья Литвинов

Лекция 4



Содержание

- flex-box
- Bootstrap
- Верстка макета



Flex-box layout

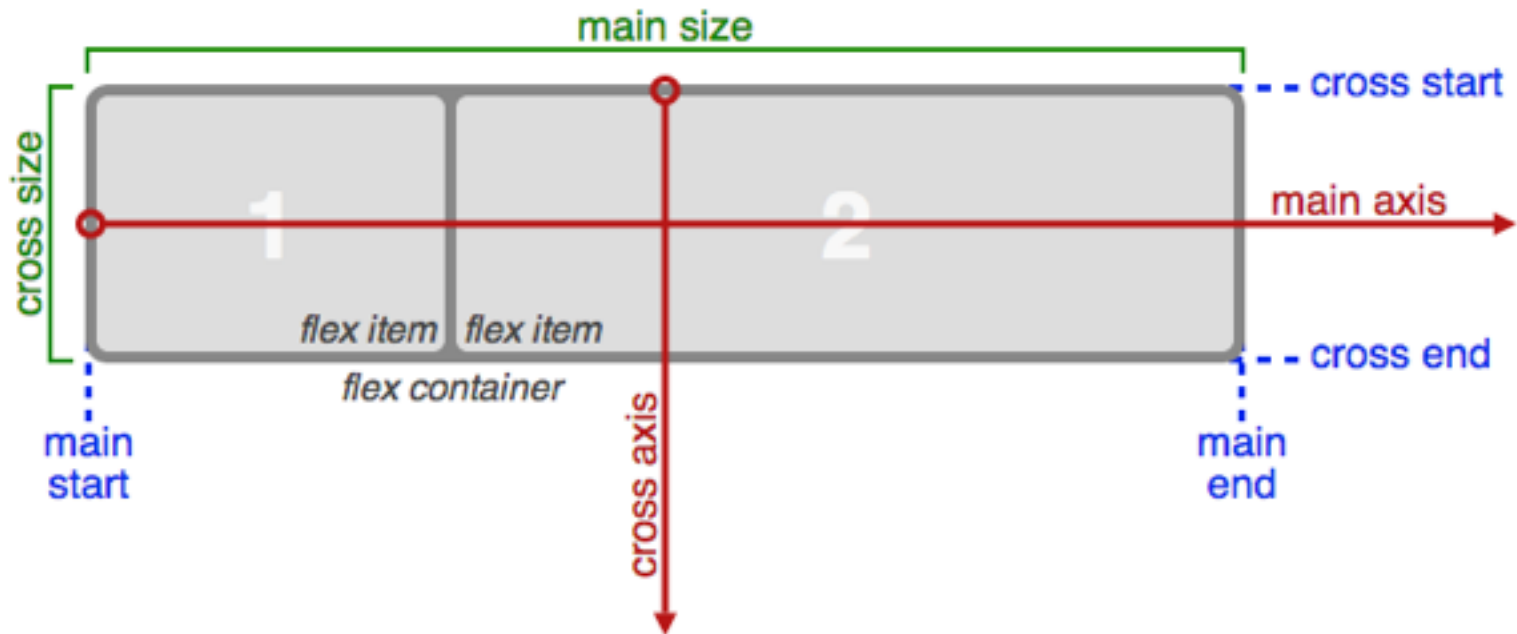
Flexbox layout

Модуль **Flexbox**-лейаута (flexible box — «гибкий блок», на данный момент W3C Candidate Recommendation) ставит задачу предложить более эффективный способ вёрстки, выравнивания и распределения свободного места между элементами в контейнере, даже когда их размер неизвестен и/или динамический (отсюда слово «гибкий»).



Flexbox layout

Flexbox — это целый модуль, а не просто единичное свойство, он объединяет в себе множество свойств. Некоторые из них должны применяться к контейнеру (родительскому элементу, так называемому flex-контейнеру), в то время как другие свойства применяются к дочерним элементам, или flex-элементам.



Flexbox layout

main-axis - главная ось, вдоль которой располагаются flex-элементы. Обратите внимание, она необязательно должна быть горизонтальной, всё зависит от свойства justify-content (см. ниже).

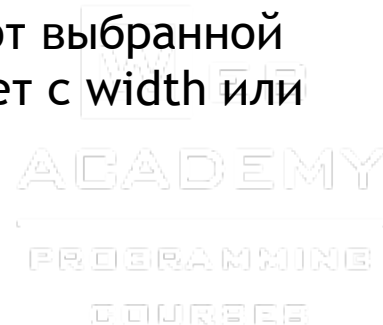
main-start | **main-end** - flex-элементы размещаются в контейнере от позиции **main-start** до позиции **main-end**.

main size - ширина или высота flex-элемента в зависимости от выбранной основной величины. Основная величина может быть либо шириной, либо высотой элемента.

cross axis - поперечная ось, перпендикулярная к главной. Её направление зависит от направления главной оси.

cross-start | **cross-end** - flex-строки заполняются элементами и размещаются в контейнере от позиции **cross-start** и до позиции **cross-end**.

cross size - ширина или высота flex-элемента в зависимости от выбранной размерности равняется этой величине. Это свойство совпадает с **width** или **height** элемента в зависимости от выбранной размерности.



Flexbox layout

Свойства родительского контейнера

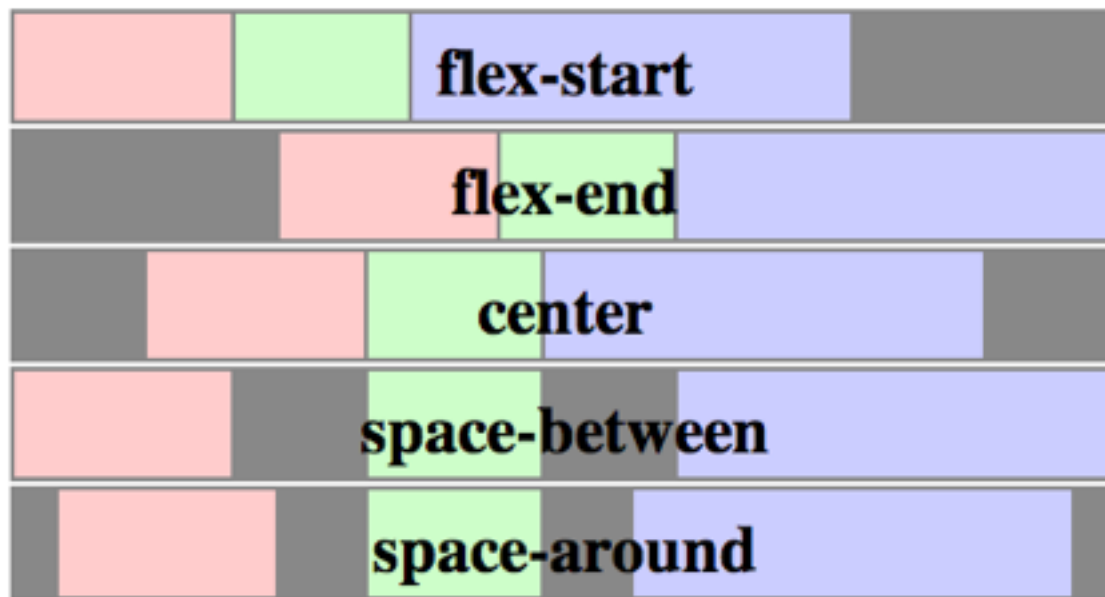
display: flex | inline-flex - определяет flex-контейнер (инлайновый или блочный в зависимости от выбранного значения)

flex-direction - устанавливает главную ось main-axis, определяя тем самым направление для flex-элементов, размещаемых в контейнере.

flex-wrap | nowrap | wrap | wrap-reverse - определяет, будет ли контейнер однострочным или многострочным, а также направление поперечной оси, определяющей направление, в котором будут располагаться новые строки.

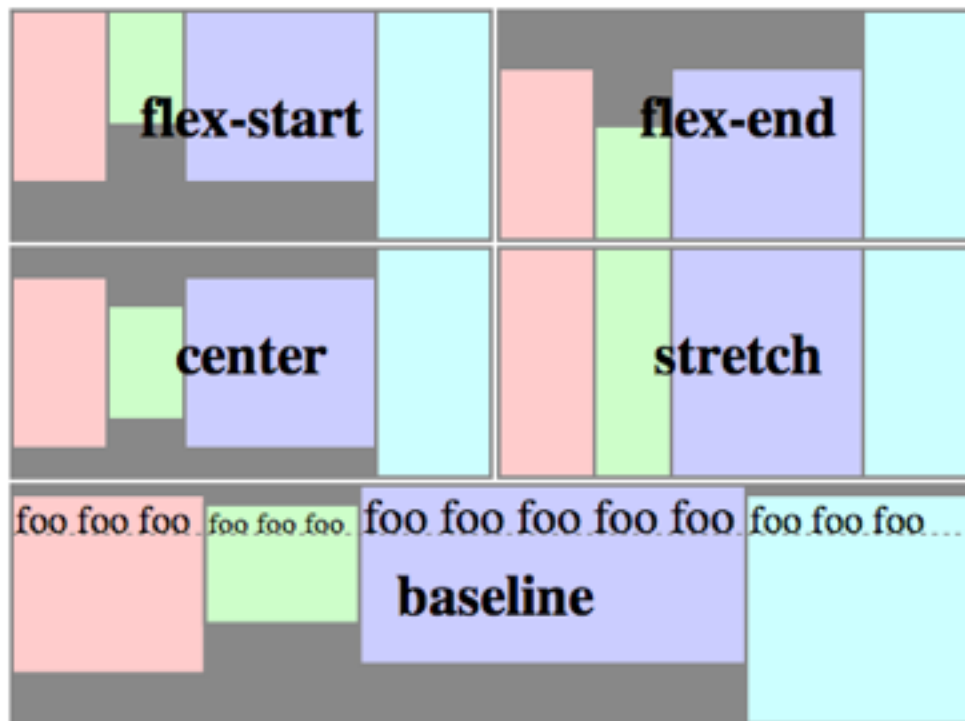
Flexbox layout

justify-content: flex-start | flex-end | center | space-between | space-around - Определяет выравнивание относительно главной оси. Помогает распределить оставшееся свободное место в случае, когда элементы строки не «тянутся», либо тянутся, но уже достигли своего максимального размера. Также позволяет в некотором роде управлять выравниванием элементов при выходе за границы строки.



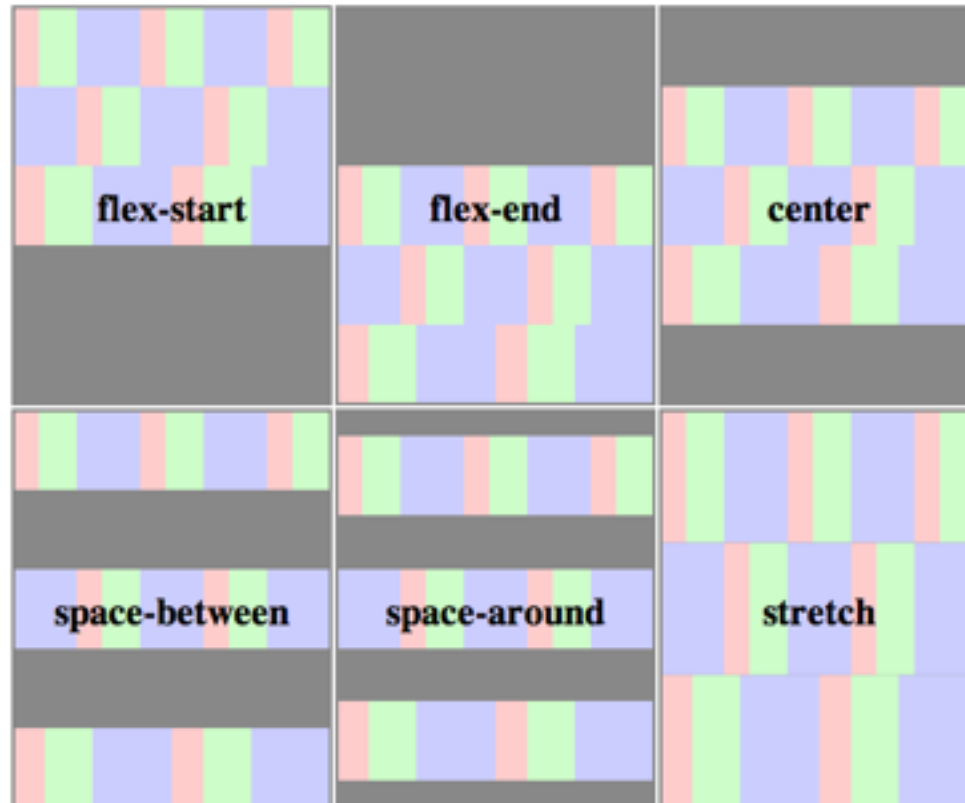
Flexbox layout

align-items: **flex-start** | **flex-end** | **center** | **baseline** | **stretch** - определяет поведение по умолчанию для того, как flex-элементы располагаются относительно поперечной оси на текущей строке. Считайте это версией justify-content для поперечной оси (перпендикулярной к основной).



Flexbox layout

align-content: flex-start | flex-end | center | space-between | space-around | stretch - выравнивает строки flex-контейнера при наличии свободного места на поперечной оси аналогично тому, как это делает justify-content на главной оси.



Flexbox layout

Свойства дочерних элементов

order - свойство `order` может управлять порядком расположения дочерних элементов в контейнере.

flex-grow: `<number>` - определяет для flex-элемента возможность «вырастать» при необходимости. Принимает безразмерное значение, служащее в качестве пропорции. Оно определяет, какую долю свободного места внутри контейнера элемент может занять. Если у всех элементов свойство `flex-grow` задано как 1, то каждый потомок получит внутри контейнера одинаковый размер. Если вы задали одному из потомков значение 2, то он заберёт в два раза больше места, чем другие.

flex-shrink: `<number>` - определяет для flex-элемента возможность сжиматься при необходимости

flex-basis: `<length>` - Определяет размер по умолчанию для элемента перед распределением пространства в контейнере

align-self - позволяет переопределить выравнивание, заданное по умолчанию или в `align-items`, для отдельных flex-элементов.

Twitter bootstrap

Что такое bootstrap?

- Bootstrap бесплатный CSS фреймворк для быстрой и легкой разработки
- Включает в себя HTML и CSS шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.

Фреймворк начал разрабатываться в качестве внутренней библиотеки компании **Twitter** под названием **Twitter Blueprint**. После нескольких месяцев разработки он был открыт под названием Bootstrap 19 августа 2011 года.



Основные компоненты

Сетки — заранее заданные размеры колонок, которые можно сразу же использовать, например ширина колонки 140px относится к классу `.span2` (`.col-md-2` в третьей версии фреймворка), который можно использовать в CSS описании документа.

Шаблоны — Фиксированный или резиновый шаблон документа.

Типографика — Описания шрифтов, определение некоторых классов для шрифтов, таких как код, цитаты и т. п.

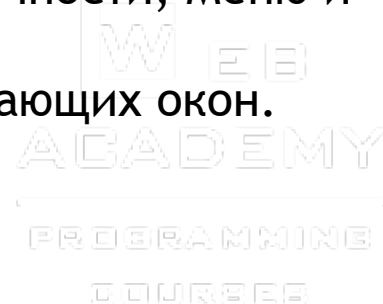
Медиа — Представляет некоторое управление изображениями и Видео.

Таблицы — Средства оформления таблиц, вплоть до добавления функциональности сортировки.

Формы — Классы для оформления форм и некоторых событий происходящих с ними.

Навигация — Классы оформления для Табов, Вкладок, Страничности, Меню и Тулбара.

Алерты — Оформление диалоговых окон, Подсказок и Всплывающих окон.



ОСНОВЫ

<http://getbootstrap.com/getting-started/#download>

```
bootstrap/  
+-- css/  
|   +-- bootstrap.css  
|   +-- bootstrap.min.css  
+-- js/  
|   +-- bootstrap.js  
|   +-- bootstrap.min.js  
+-- img/  
|   +-- glyphs-halflings.png  
|   +-- glyphs-halflings-white.png  
L-- README.md
```

Обзор возможностей

КОНЕЦ