

ДЕТСКАЯ ОНЛАЙН-ШКОЛА ПРОГРАММИРОВАНИЯ

HELLO WORLD



#ШПАРГАЛОЧКИ

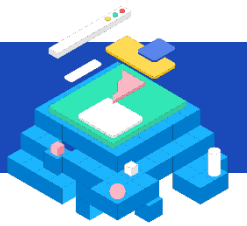
СОЗДАНИЕ САЙТОВ FRONT-END РАЗРАБОТКА

Начальный уровень

Материалы подготовлены отделом методической разработки

Больше полезных материалов и общения в нашем комьюнити
в Telegram: https://t.me/hw_school





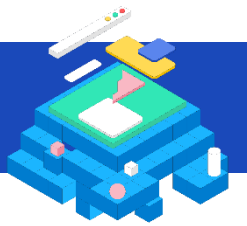
Сайт арт-студии часть 2: flex-shrink, flex-basis, flex, order



Знакомимся ближе с flex-grow

flex-grow умеет не только указывать, сколько пространства элемент может занять внутри родителя, но задавать **пропорции элементов относительно друг друга**.

Так, если всем элементам флекс-контейнера свойству **flex-grow** задано значение **1**, они будут одинакового размера. А если у одного элемента **flex-grow: 1**, а у второго **flex-grow: 2**, то второй элемент будет в 2 раза больше первого (при наличии в контейнере свободного места для роста).



flex-basis

flex-basis - свойство, влияющее на размер элементов флекс-бокса. Задается именно элементам, отвечает за **размер вдоль главной оси**.

flex-basis принимает значения в разных единицах измерения, включая **px** и **%**:

```
.item {
```

```
    flex-basis: 350px; /* размер элемента вдоль главной оси */
```

```
}
```



order

Свойство **order** пригодится, когда элементам flex-контейнера нужно **изменить порядок**, не меняя при этом их расположение в html-коде.

Достаточно задать каждому элементу уникальный класс и прописать свойство **order** с порядковыми номерами элементов.

По умолчанию у всех элементов в **order** стоит **0**. Поэтому, если переставить нужно только один элемент, можно прописать ему отрицательное значение (**order: -1**) - тогда он сдвинется **на первое место**. Или любое число **больше 0** - тогда он сдвинется **в конец**. Разумеется, если свойство **order** меняется у нескольких элементов, нужно следить, чтобы указанное число соответствовало желаемому расположению.



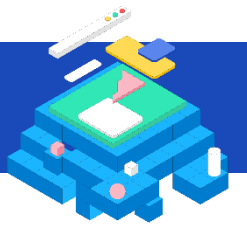
flex-shrink

Если flex-элемент нужно увеличить, можно использовать свойство flex-grow. А если размер нужно наоборот уменьшить, отлично подойдет еще одно свойство - **flex-shrink**.

flex-shrink работает очень похоже на **flex-grow**: ему нужно передать число, означающее, на сколько по отношению к остальным элементам будет **уменьшен** текущий элемент.

.item1 {	.item2 {
width: 100px;	width: 100px;
flex-shrink: 1;	flex-shrink: 2;
}	}

В этом примере, если контейнер с .item1 и .item2 меньше 200px в ширину, то оба элемента должны будут сжаться в **соотношении 2:1**. Например, если контейнер будет уменьшен до **100px**, первый элемент займет **66.6px**, а второй - **33.3px**.



flex

flex - это сокращенное свойство, позволяющее в одну строку записать значения **flex-grow**, **flex-shrink** и **flex-basis**:

```
.item {  
    flex: 1, 2, 300px;  
}
```

Этот элемент получит размер **flex-basis: 300px** вдоль основной оси, а также у него установлены свойства **flex-grow: 1** и **flex-shrink: 2**.