**display: flex –** блоки сжались и выстроились в одну строчку

displayМногоцелевое свойство, которое определяет, как элемент должен быть показан в документе.

flex Элемент ведёт себя как блочный и выкладывает содержимое согласно флекс-модели.

**flex-wrap** (перенос) **–** перенос блоков внутри контейнера при необходимости

**display: flex**

**flex-wrap: wrap**

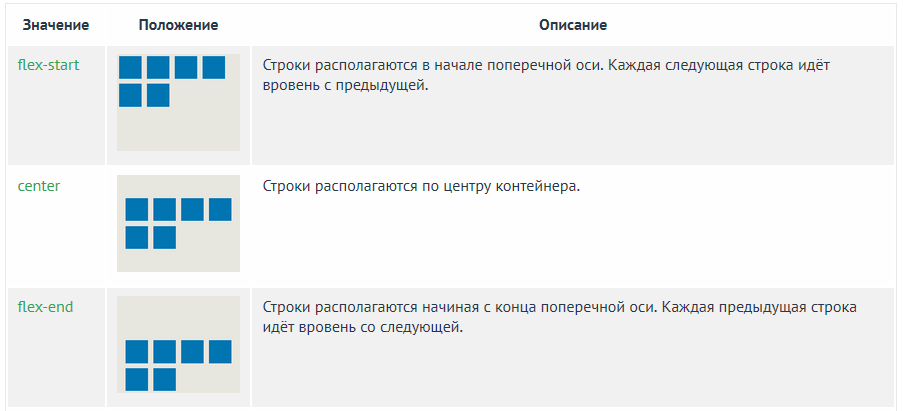
**justify-content** (оправдывать) **-** распределяет пространство вокруг флекс-элементов вдоль главной оси контейнера.

**display: flex**

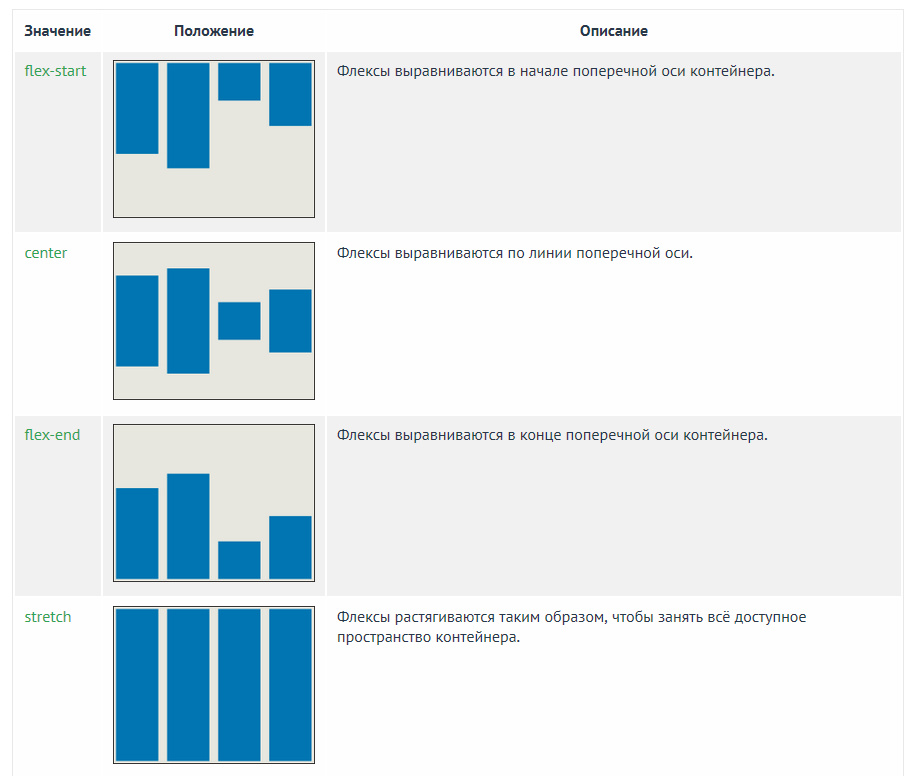
**flex-wrap: wrap**

**justify-content: center;**

**align-content (выравнивать) -** выравнивания строк внутри флекс-контейнера по поперечной оси при наличии свободного пространства.



**align-items** (выравнивать элементы) - выравнивает флекс-элементы внутри контейнера в перпендикулярном направлении.



**.body {**

**display: flex;**

**justify-content: center;**

**align-items: center;**

**}**

**Трасформация**

**Transform -** трансформирует элемент, в частности, позволяет его масштабировать, вращать, сдвигать, наклонять, а также комбинировать виды трансформаций, перечисляя функции трансформации через пробел.

Делаем при помощи псевдокласса **:active**

**Повороат**

**rotate -** поворачивает элемент в двумерном пространстве на заданный угол относительно точки трансформации

**block:active {**

**transform: rotate(45deg);**

**}**

**Масштабирование**

**scaleX** - масштабирует элемент по горизонтали.

**scaleY** - масштабирует элемент по вертикали.

**.block:active {**

**transform: scaleX(1.5) scaleY(0.5);**

**}**

**Прекос**

**skewX** - наклоняет элемент на заданный угол по вертикали.

**skewY** - наклоняет элемент на заданный угол по горизонтали.

**.block:active {**

**transform: skewX(30deg) skewY(40deg);**

**}**

**Перемещение**

**Translate** - сдвигает элемент на заданное значение по горизонтали и вертикали.

**.block:active {**

**transform: translate(40px, -20px);**

**}**

**Переходы**

**transition** - устанавливает эффект перехода между двумя состояниями элемента, они могут быть определены с помощью псевдокласса **:hover** или **:active**, а также динамически через **JavaScript**.

**.block {**

**background: linear-gradient(45deg, orange, red);**

**border-radius: 5px;**

**width: 100px;**

**height: 100px;**

**transition: transform 1s ease,**

**width 1s linear 1.5s;**

**}**

**.block:hover {**

**transform: rotate(720deg);**

**width: 300px;**

**}**

**Селектор дочерних элементов** – действует только на дочерние элементы

**селектор-1 > селектор-2 {**

**…**

**}**