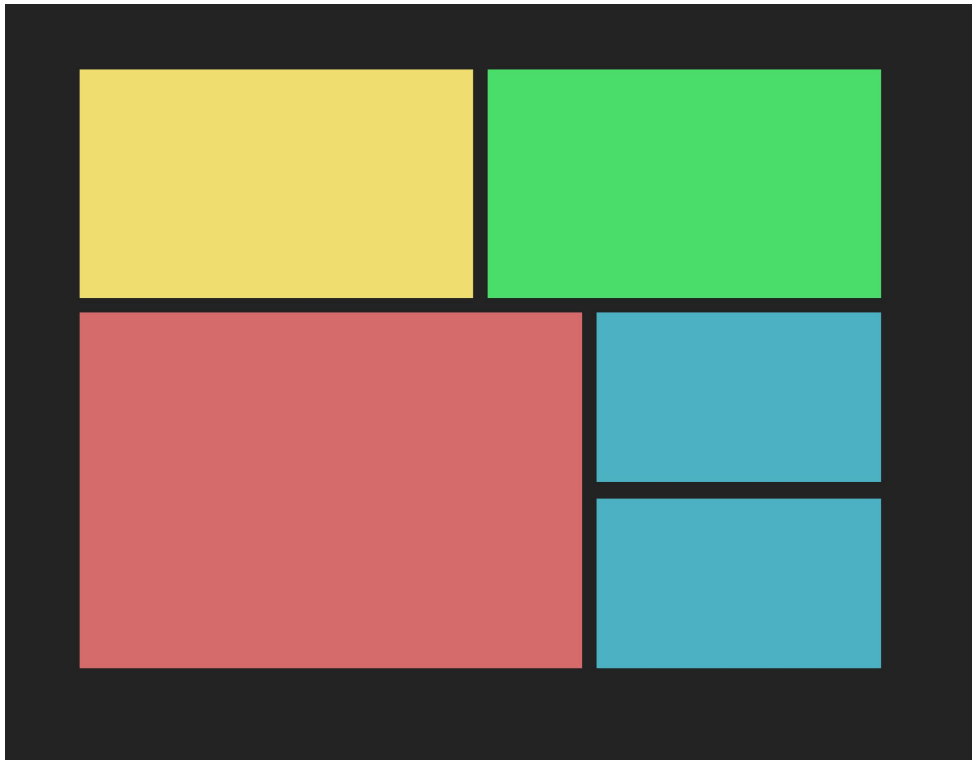
Flex-box

Цель Темы 1:
Ознакомление с главной и дочерней осью.
Выучить основные команды для простого позиционирования контента на странице.

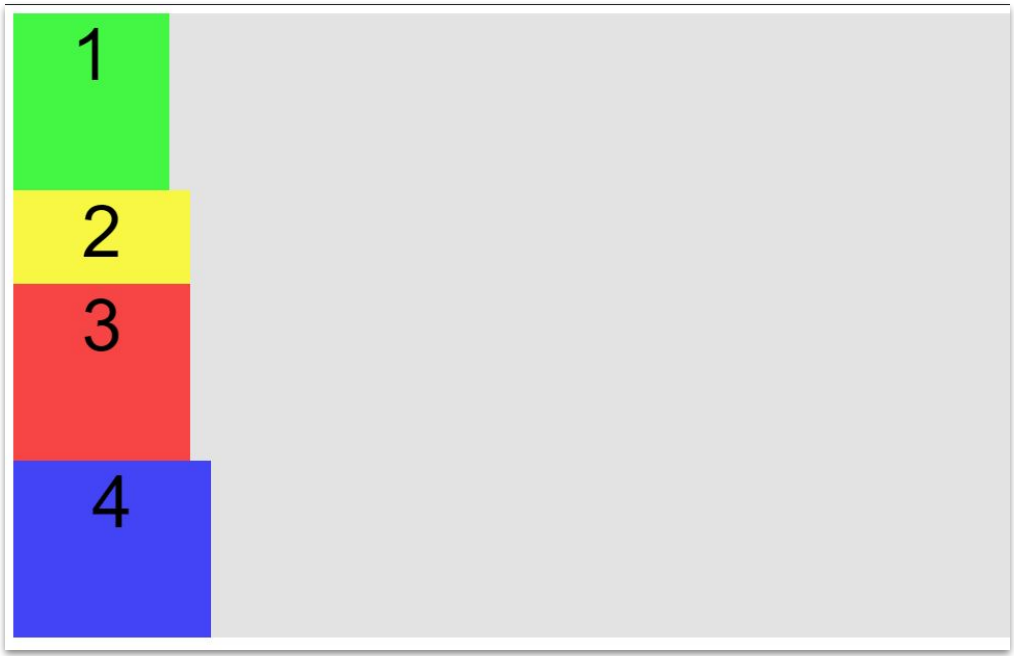
Flex-box - система компоновки элементов на WEB странице, которая позволяет быстро и легко **позиционировать** контент на сайте без лишних усилий



Задача 1

Реализовать пример справа.
Структура:

```
div-контейнер
    div
    div
    div
    div
/div-контейнер
```



Пример оформления блока:

```
.one {
    background: rgb(0, 255, 0, 0.7);
    width: 150px;
    height: 170px;
    font-family: Helvetica;
    font-size: 70px;
    text-align: center;
}
```

Необходимо применить ко всем остальным блоками.

Решение

```html
<!DOCTYPE html>
<html>
<head>
    <title>Какое-то название страницы</title>
    <link rel="stylesheet" type="text/css" href=
    "style.css">
</head>
<body>
    <div class="container">
        <div class="one">1</div>
        <div class="two">2</div>
        <div class="three">3</div>
        <div class="four">4</div>
    </div>
</body>
</html>
```

```css
.container {
    background: rgb(200, 200, 200, 0.5);
}
.one {
    background: rgb(0, 255, 0, 0.7);
    width: 150px;
    height: 170px;
    font-family: Helvetica;
    font-size: 70px;
    text-align: center;
}

.two {
    background: rgb(255, 255, 0, 0.7);;
    width: 170px;
    height: 90px;
    font-family: Helvetica;
```

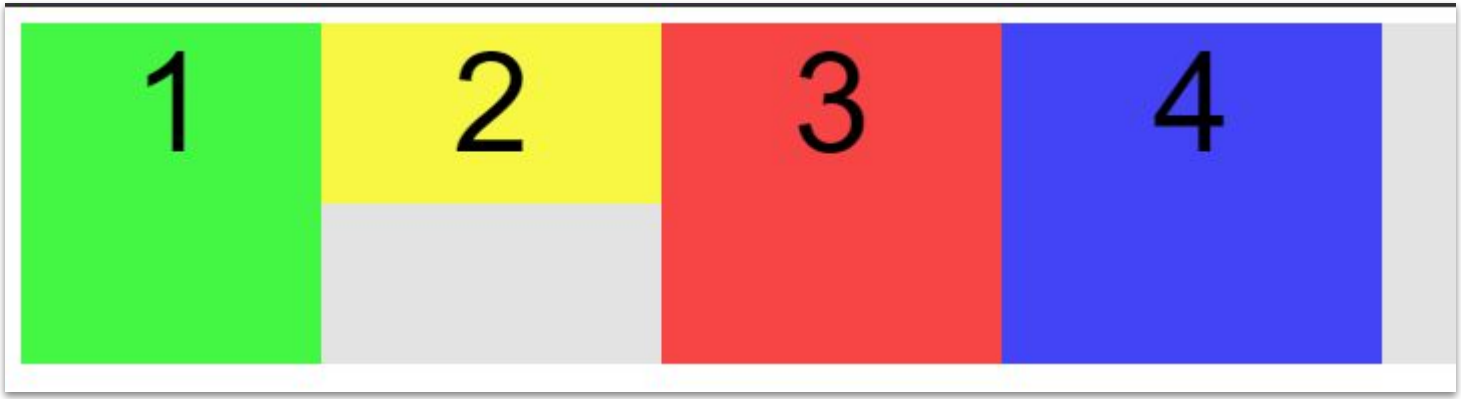
Flex-контейнер

Чтобы начать работу с flex-box, необходимо наш главный **div** сделать **flex-контейнером**.

Для этого, поменяйте свойство **display** у главного **div**-а**,** и задайте ему значение **flex:**

```
.container {
    background: rgb(200, 200, 200, 0.5);
    display: flex;
}
```

Результат:

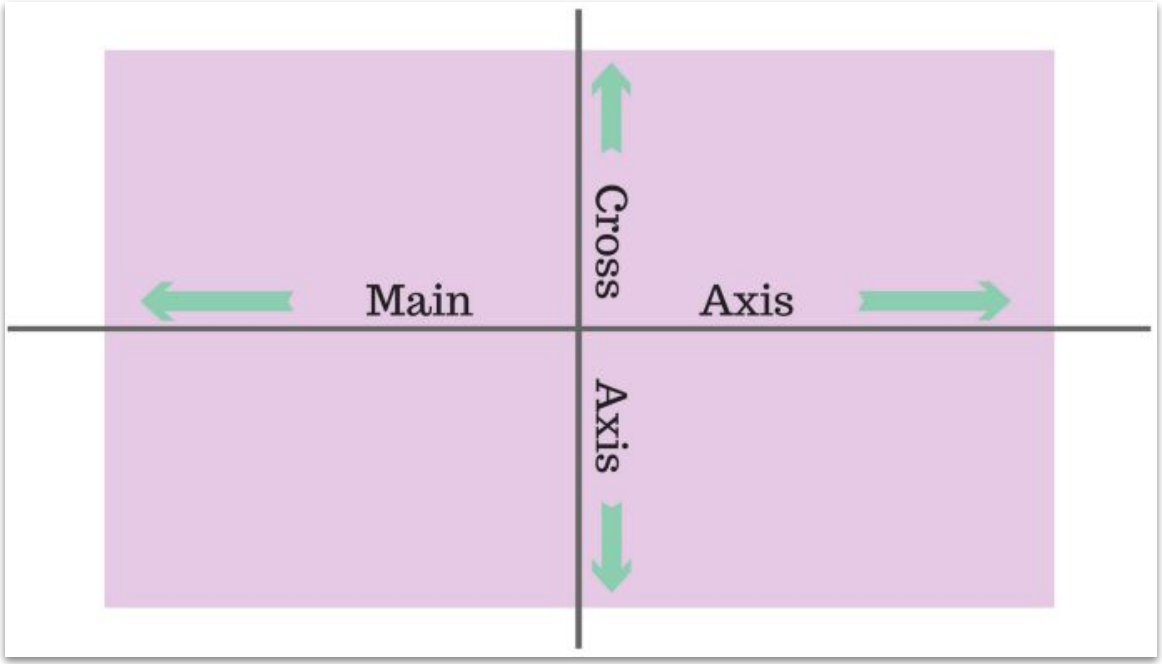


Теперь дочерние **div-**ы (1, 2, 3, 4) поддаются законам **flex-box** и позволяют позиционировать себя внутри **div**-а родителя очень просто и понятно.

Для этого необходимо запомнить лишь несколько простых свойств и значений:

Flex-direction

У flex-контейнера есть две оси: главная ось и перпендикулярная ей.



По умолчанию все предметы располагаются вдоль главной оси: слева направо.

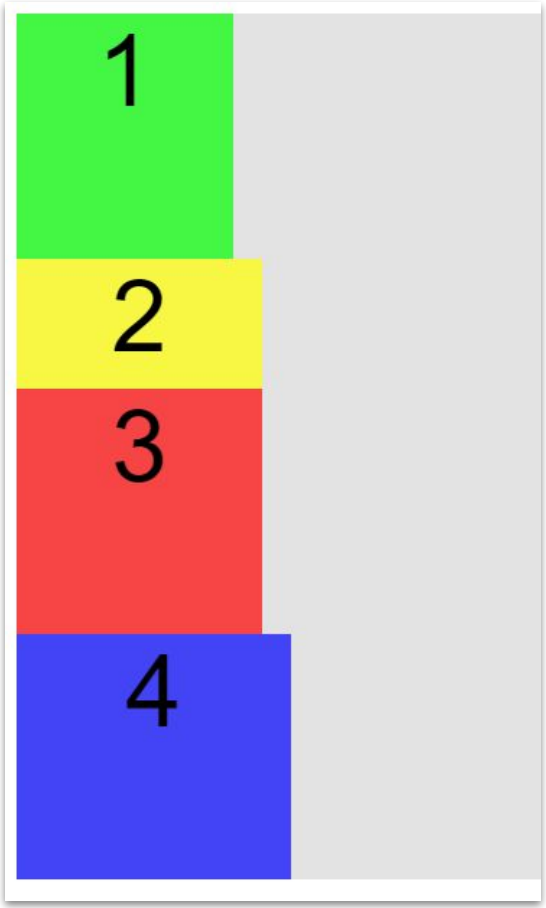
Поэтому наши квадраты выровнялись в линию, когда мы применили
display: flex.
Однако **flex-direction** позволяет вращать главную ось:

```
.container {
    background: rgb(200, 200, 200, 0.5);
    display: flex;
    flex-direction: column;
}
```

Результат:

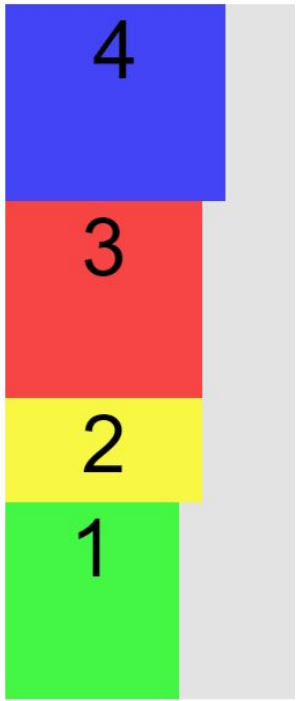
Важно заметить, что **flex-direction: column** не выравнивает квадраты по оси, перпендикулярной главной. Главная ось сама меняет свое расположение и теперь направлена сверху вниз.

Чтобы все вернуть, необходимо поменять свойство **flex-direction: row;**



Есть еще парочка свойств для **flex-direction:**

column-reverse



row-reverse

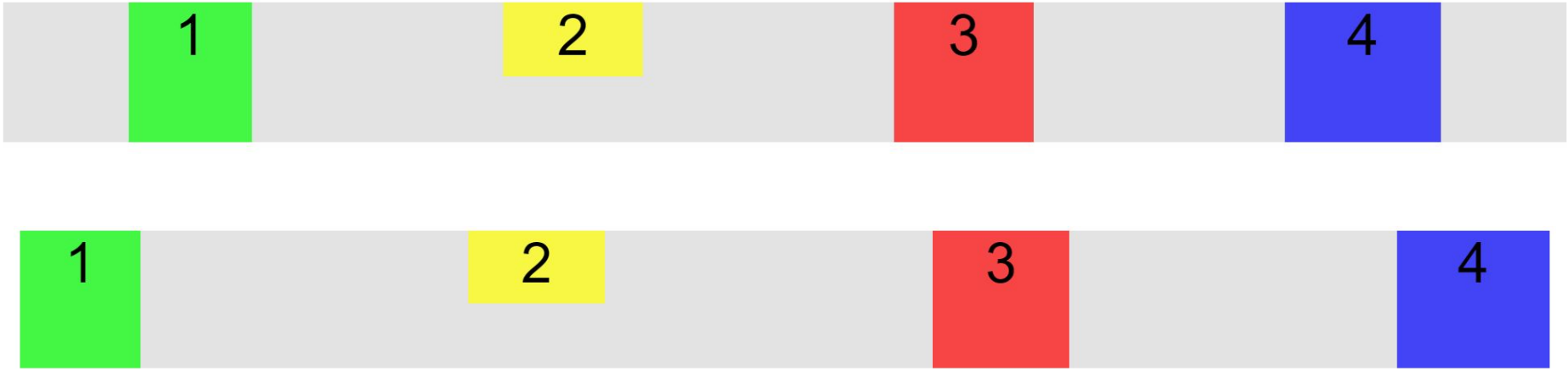
Justify-content

Justify-content отвечает за выравнивание элементов по главной оси.

Это свойство может принимать 5 значений:

1. flex-start;
2. flex-end;
3. center;
4. space-between;
5. space-around.

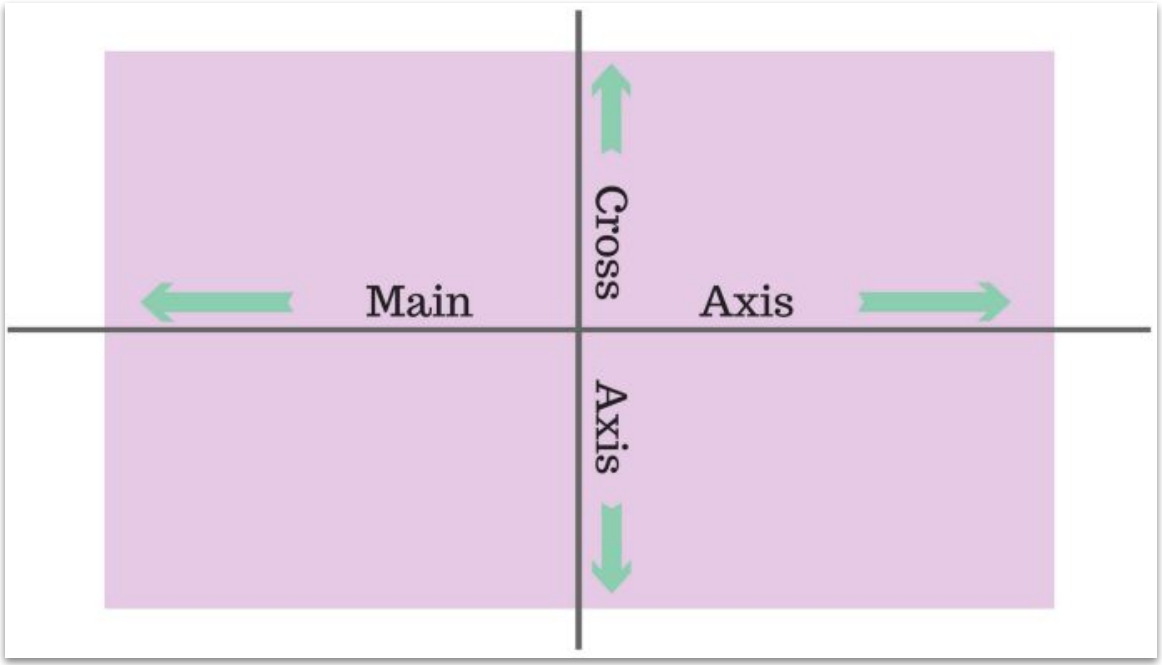
Опробуйте **каждое** из значений в режиме **flex-direction: row.**
Особенно обратите внимание на 2 последних. В чем между ними разница?



Space-between (2) задает одинаковое расстояние между квадратами, но не между контейнером и квадратами. **Space-around(2)** также задает одинаковое расстояние между квадратами, но теперь расстояние между контейнером и квадратами равно половине расстояния между квадратами.

Align-items

Если **justify-content** работает с главной осью, то **align-items** работает с осью, перпендикулярной главной оси.



Вернемся обратно к **flex-direction: row** и пройдемся по командам **align-items:**

Это свойство может принимать 5 значений:

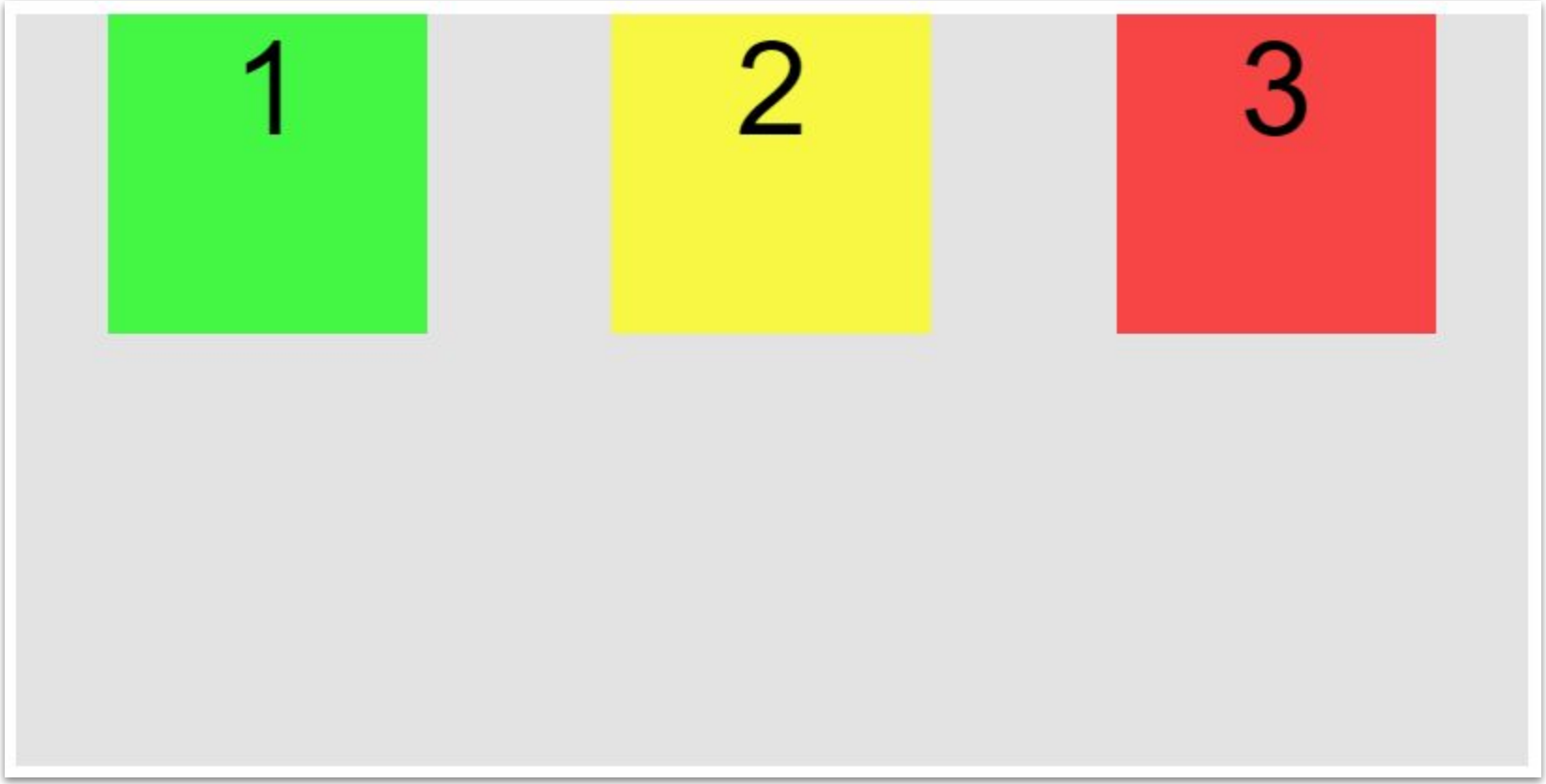
1. flex-start;
2. flex-end;
3. center;
4. stretch;
5. baseline;

Опробуйте **каждое** из значений в режиме **flex-direction: row.**
В чем разница между flex-start и baseline?

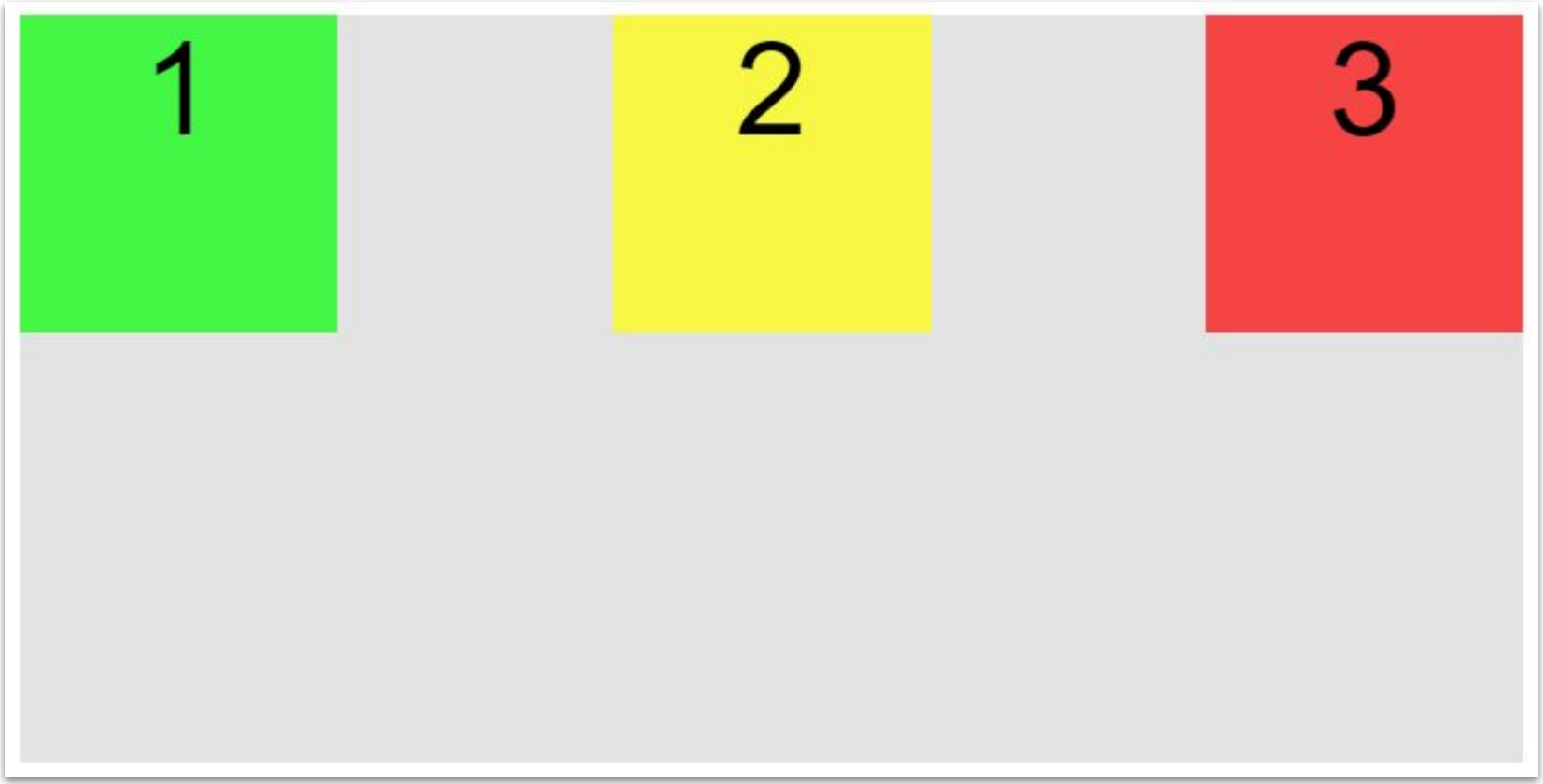
Домашнее задание #1

Домашнее задание #2

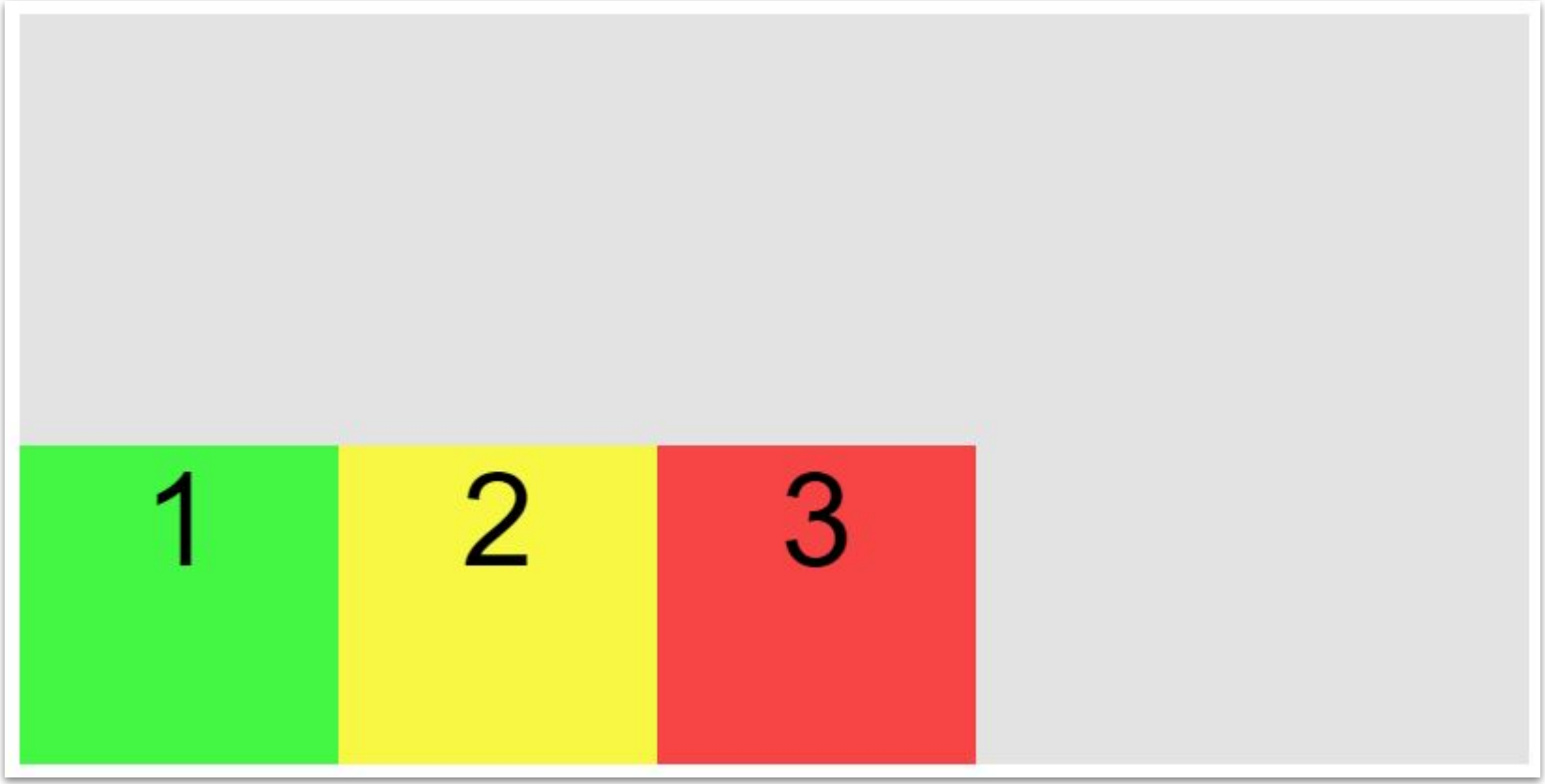
Домашнее задание #3



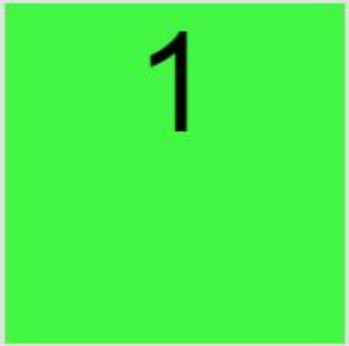
Домашнее задание #4



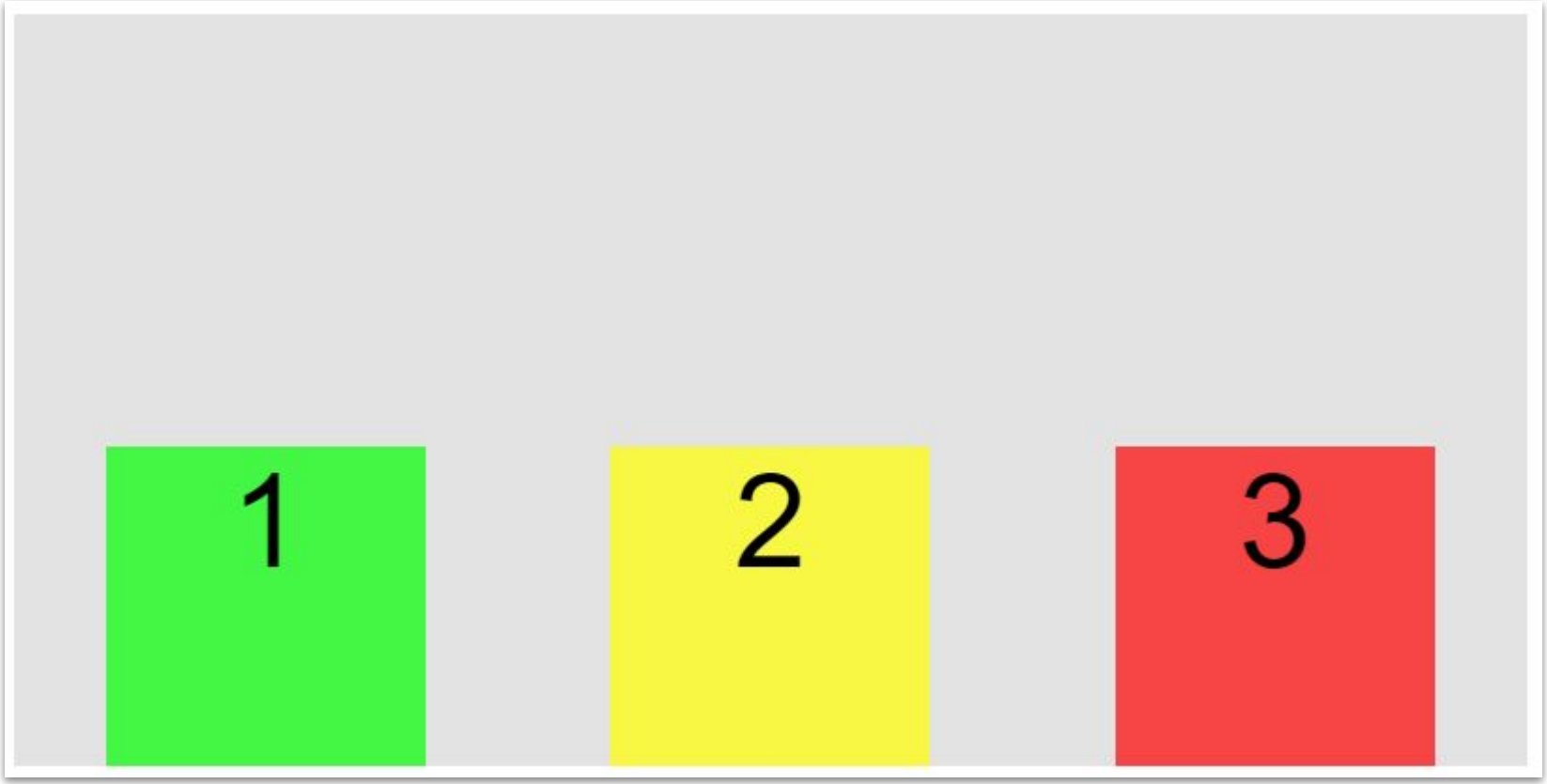
Домашнее задание #5



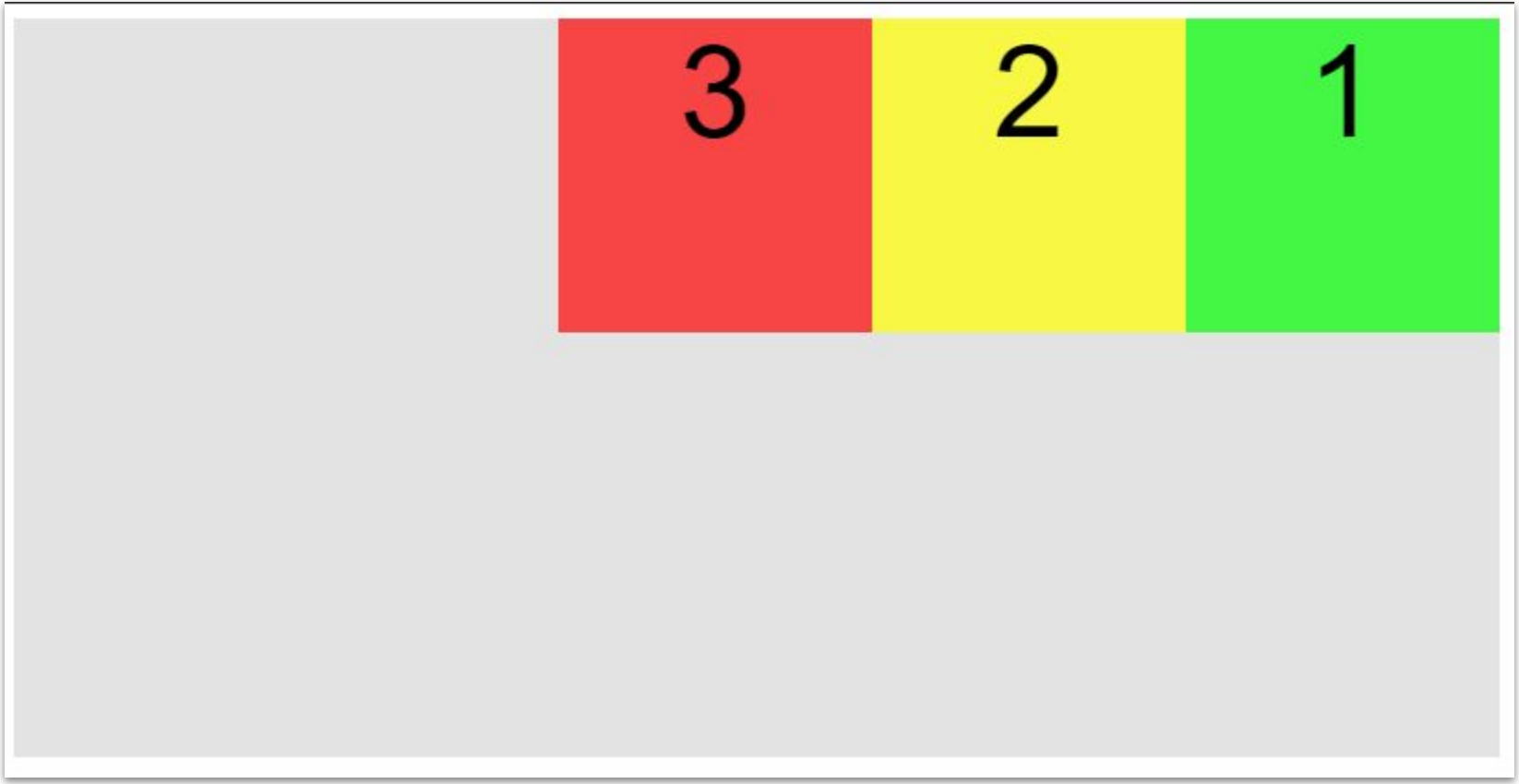
Домашнее задание #6



Домашнее задание #7



Домашнее задание #8



Домашнее задание #9