МШП 20.35 УНИВЕРСИТЕТ КОД БУДУЩЕГО Минцифры России НАЦИОНАЛЬНЫЕ ПРОЕКТЫ РОССИИ ЦИФРОВАЯ ЭКОНОМИКА

Fullstack-разработка | Занятие №10

CSS
Flexbox

МШП

Цели урока:

1. Познакомиться с моделью flexbox
2. Научиться создавать гибкие элементы
3. Создать простую сетку

МШП

Повторение

Разбор домашнего задания

МШП

Теория

Модель Flexbox

МШП

Вспомним

1. **Блочная модель** позволяет настроить размер, отступы, рамку

2. Она хорошо работает для **блочных** и **строчно-блочных** элементов

МШП

display

```
span {
    width: 200px;
    height: 100px;
    display: block;
    background-color: #eeeeee;
}
```

Свойство **display: block** делает элементы **блочными**

display

```
span {
    width: 200px;
    height: 100px;
    display: inline-block;
    background-color: #eeeeee;
}
```

Эта магическая строчка превращает любой элемент в **строчно-блочный**

МШП

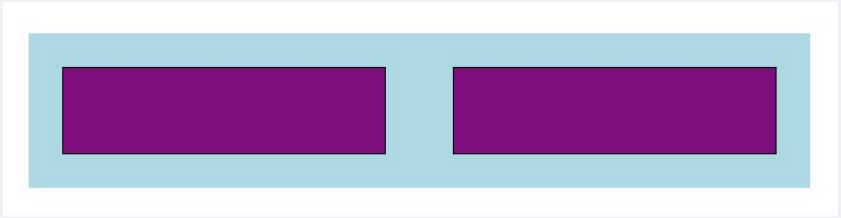
display

```
div {
    width: 200px;
    height: 100px;
    display: flex;
    background-color: #eeeeee;
}
```

Сегодня мы научимся превращать элементы во **flex-контейнеры** с помощью нового заклинания

МШП

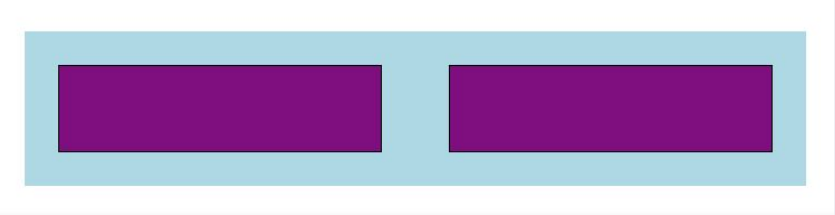
Представим типичную ситуацию: нужно сделать два блока одинаковой ширины внутри контейнера



МШП

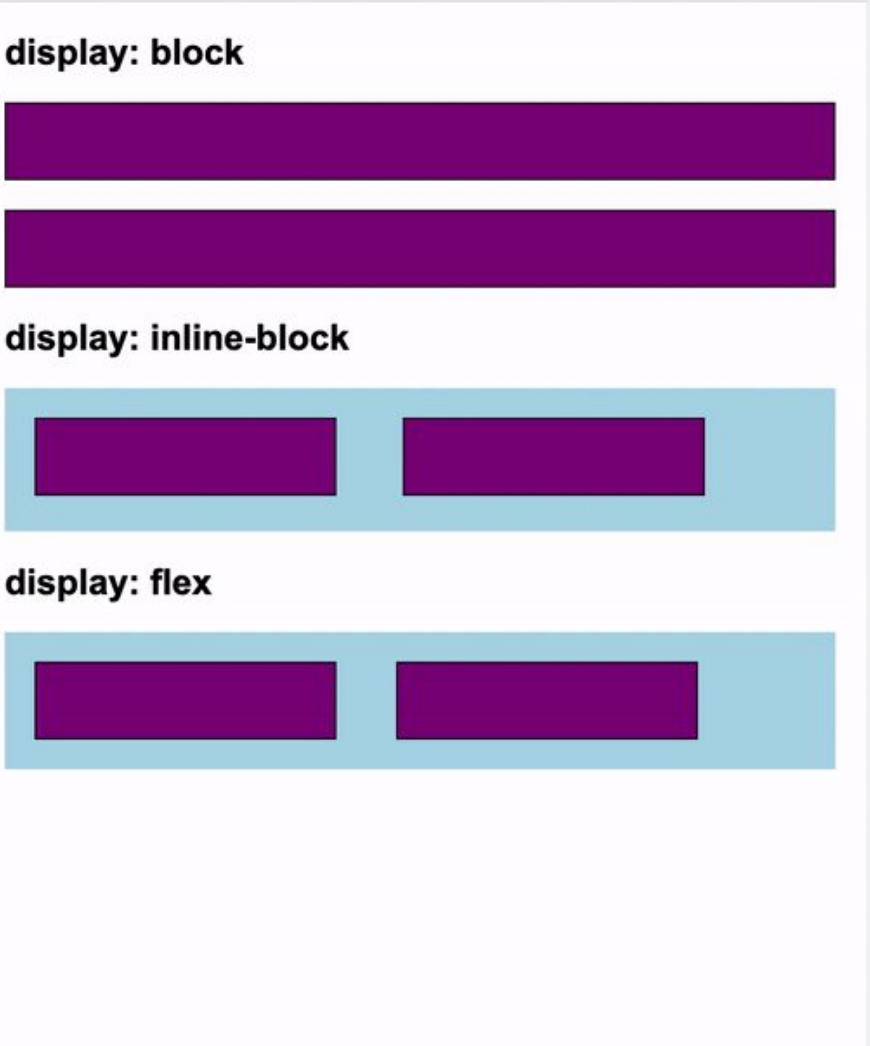
HTML

```
<div class="container">
    <div class="item"></div>
    <div class="item"></div>
</div>
```



МШП

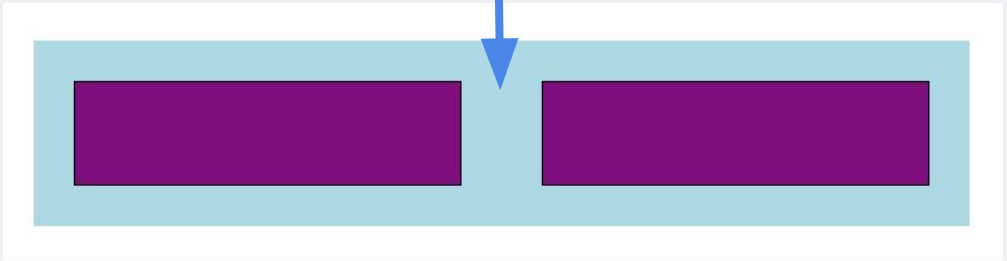
Какой вариант лучше?



МШП

flex-контейнер

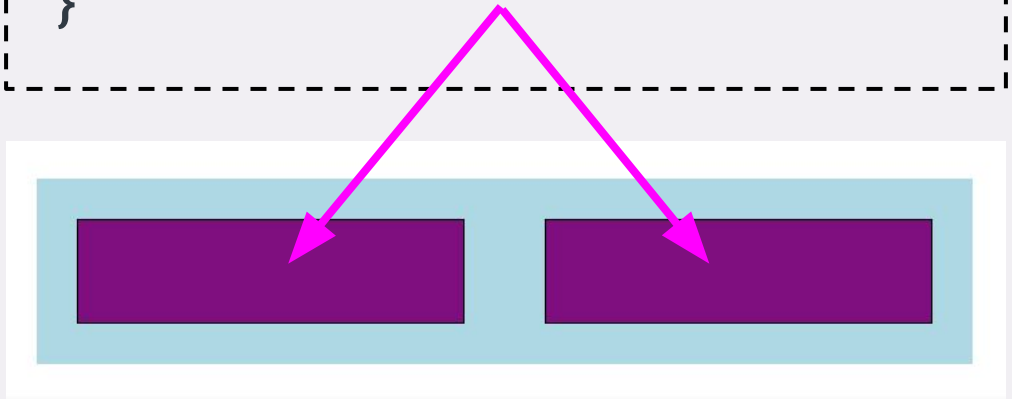
```
.container {
    display: flex;
}
```



МШП

flex-элементы

```
.item {
    width: 200px;
    height: 50px;
    margin: 20px;
}
```



МШП

Модель Flexbox

Flex-контейнер управляет **flex-элементами** внутри него

Он меняет их:

1. **положение**
2. **размер**
3. **внешние отступы**

МШП

Теория

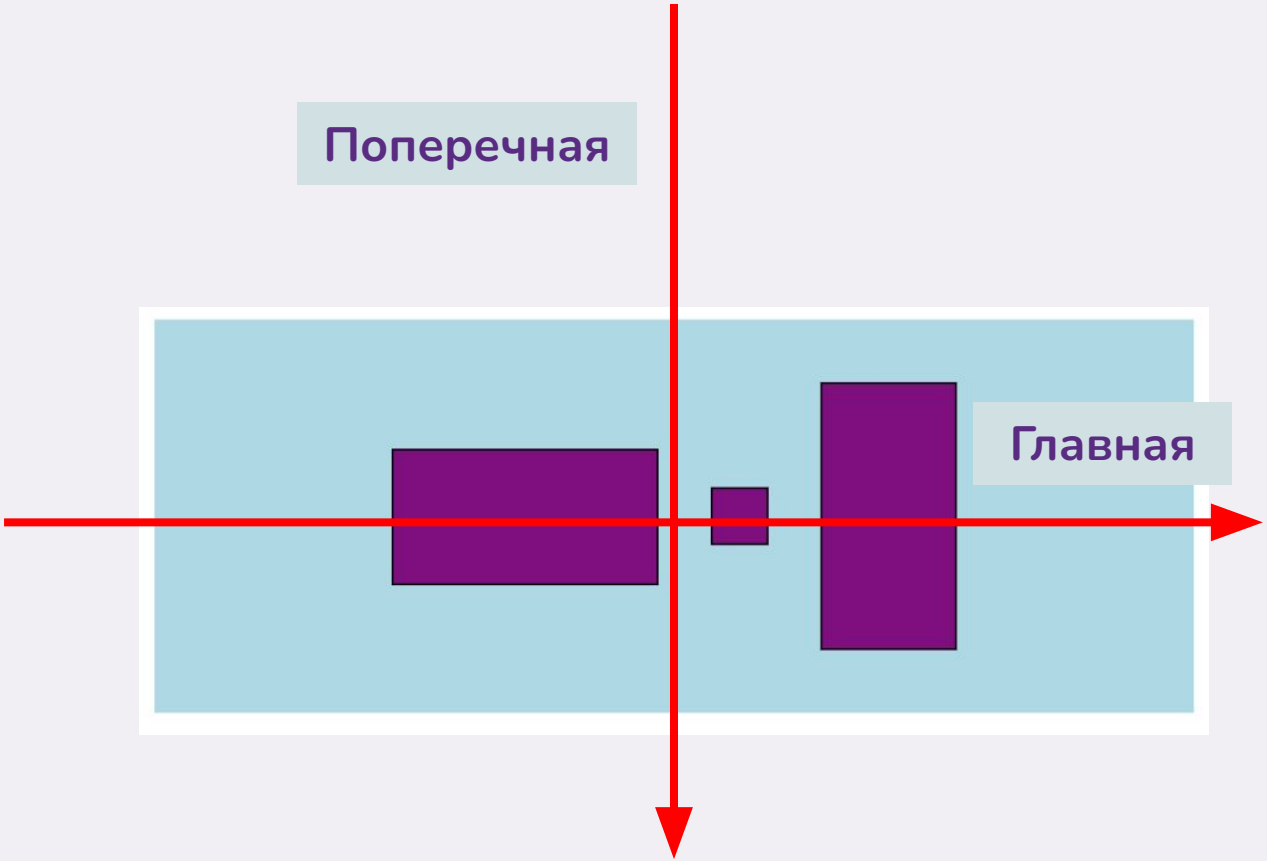
Выравнивание по осям

МШП

Внутри flex-контейнера
есть две оси, вдоль которых
можно двигать элементы

1. **Главная**

2. **Поперечная**


Поперечная
Главная

МШП

justify-content

главная ось

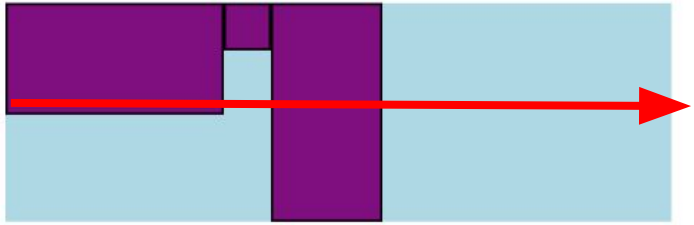
```
.container {
  display: flex;
  justify-content: flex-start;
}
```

Свойство **justify-content** управляет положением элементов вдоль главной оси

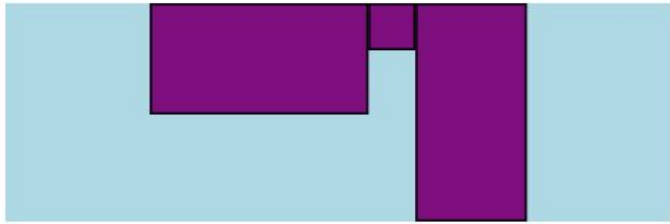
По умолчанию они прижимаются к её **началу**

МШП

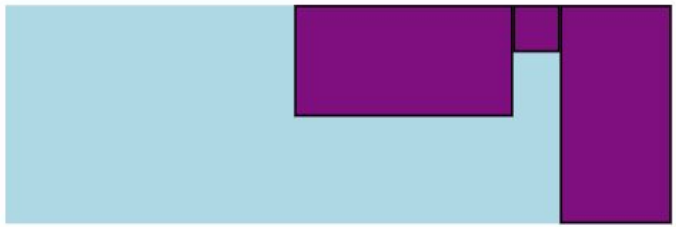
justify-content: flex-start



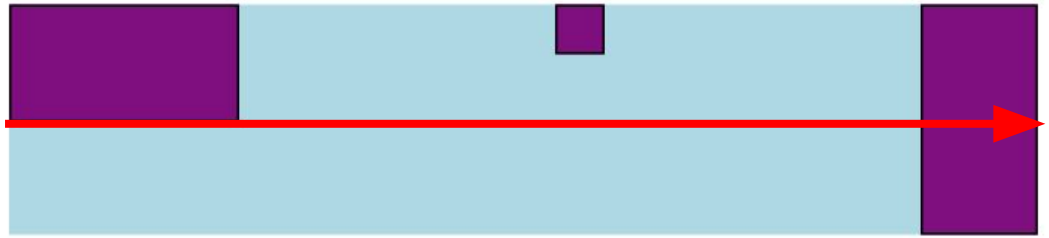
justify-content: center



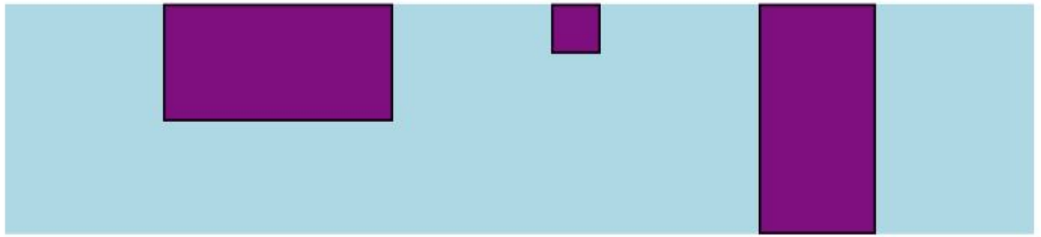
justify-content: flex-end



justify-content: space-between



justify-content: space-evenly



МШП

align-items

поперечная ось

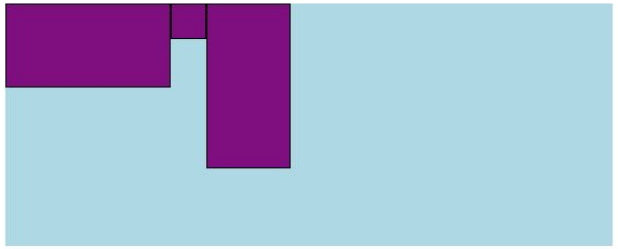
```
.container {
  display: flex;
  align-items: flex-start;
}
```

Свойство **align-items** управляет положением элементов вдоль поперечной оси

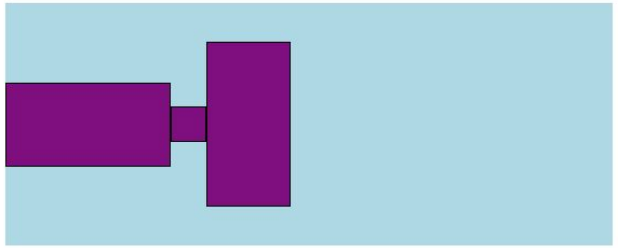
По умолчанию они прижимаются к её **началу**

МШП

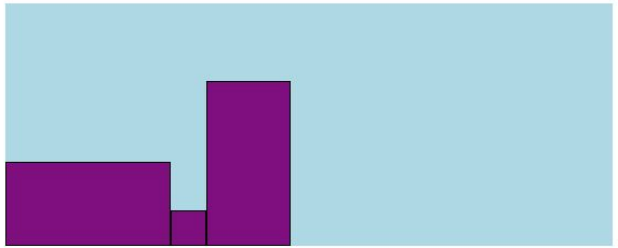
align-items: flex-start



align-items: center

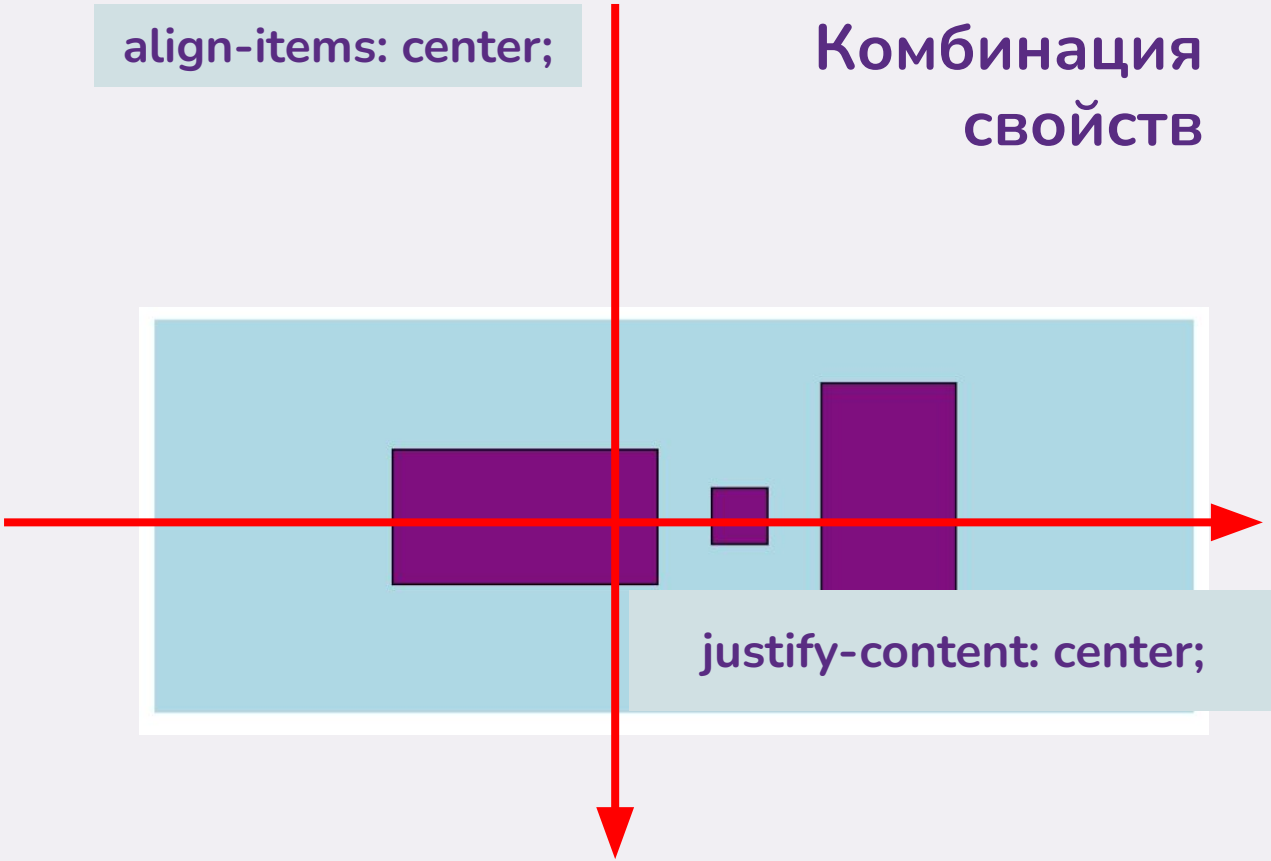


align-items: flex-end



МШП

Комбинация свойств



МШП

Практика

Знакомство с Flexbox

МШП

Теория

Ширина элементов

Модель Flexbox

Flex-контейнер управляет **flex-элементами** внутри него

Он меняет их:

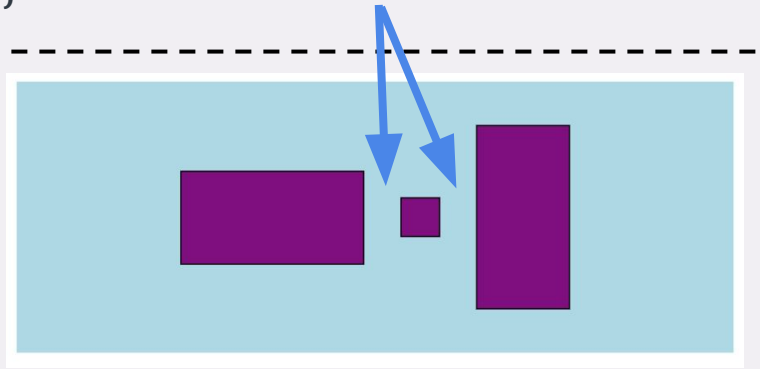
1. **положение**
2. **размер**
3. **внешние отступы**

МШП

gap

расстояние между элементами

```
.container {
  display: flex;
  align-items: center;
  justify-content: center;
  gap: 20px;
}
```



МШП

Модель Flexbox

Flex-контейнер управляет **flex-элементами** внутри него

Он меняет их:

1. **положение**
2. **размер**
3. **внешние отступы**

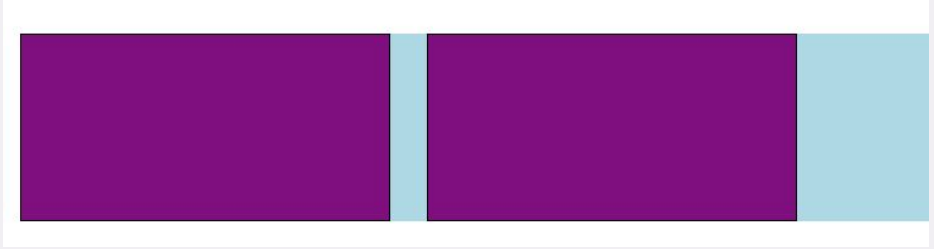
МШП

flex-basis

размер вдоль главной оси

```
.item {
    flex-basis: 200px;
    height: 100px;
}
```

flex-basis = width

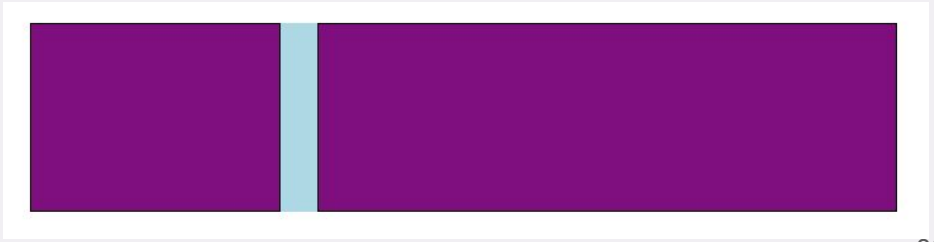


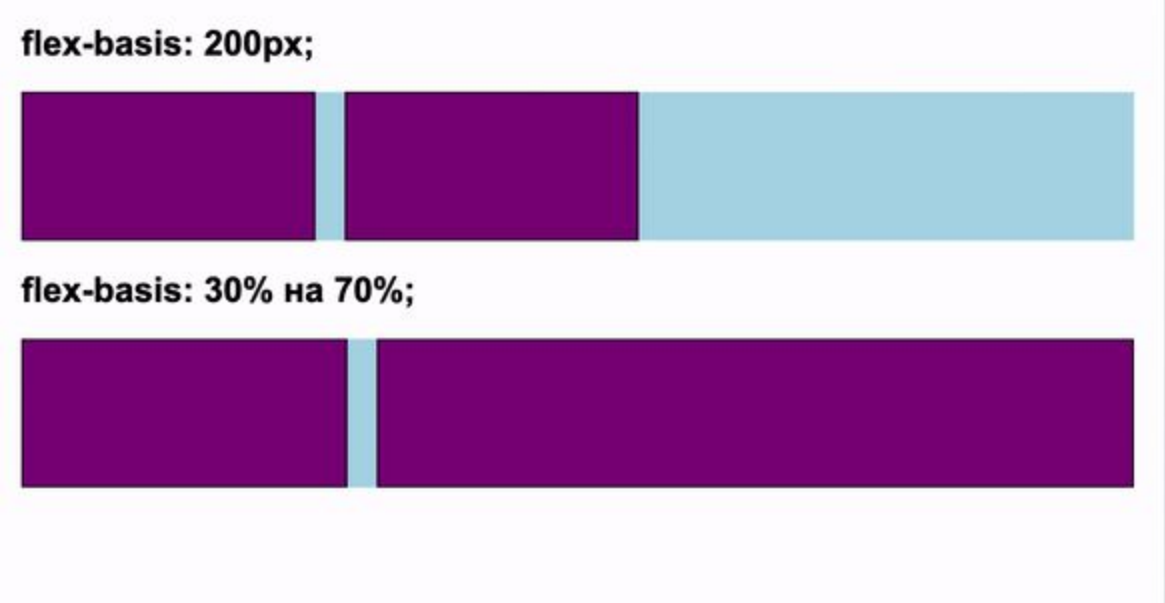
МШП

flex-basis

размер вдоль главной оси

```
.item1 {
    flex-basis: 30%;
}
.item2 {
    flex-basis: 70%;
}
```

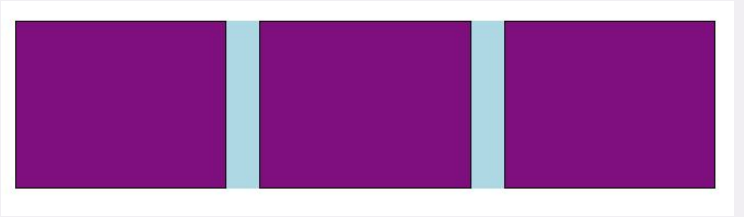


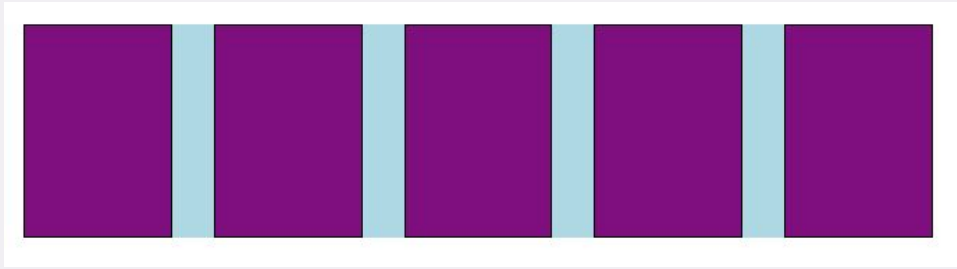

flex-basis: 200px;
flex-basis: 30% на 70%;

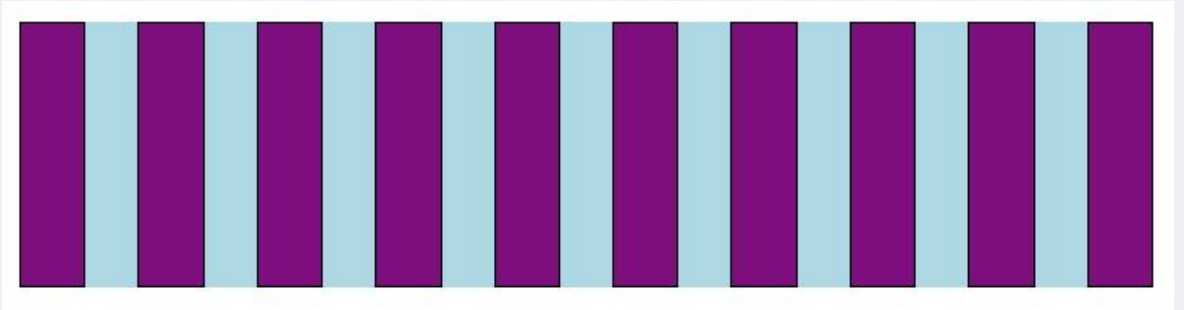
МШП

Чему равняется flex-basis?

А можно без математики????





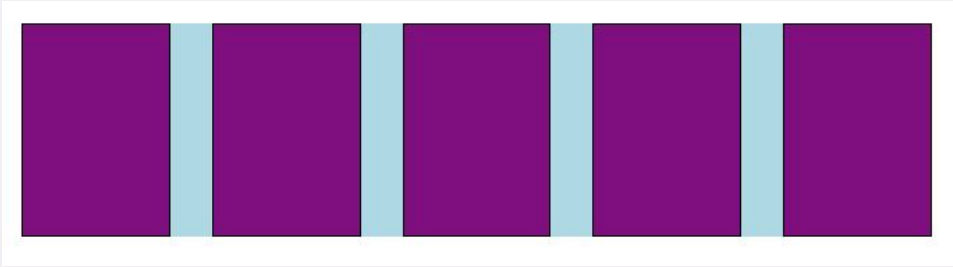


МШП

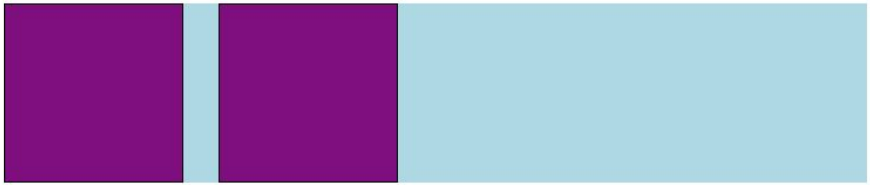
flex-grow

вытяжение вдоль главной оси

```
.item {
    flex-grow: 1;
    height: 100px;
}
```

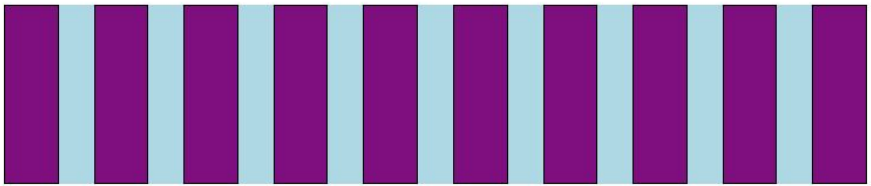


flex-grow: 0



flex-grow: 0 | flex-grow: 1

flex-grow: 1



Итоги

Контейнер

1. **gap: 20px** задаёт расстояние между элементами

Элемент

2. **flex-basis** задаёт размер вдоль главной оси
3. **flex-grow: 1** включает вытяжение вдоль главной оси

МШП

Практика

Резиновые колонки

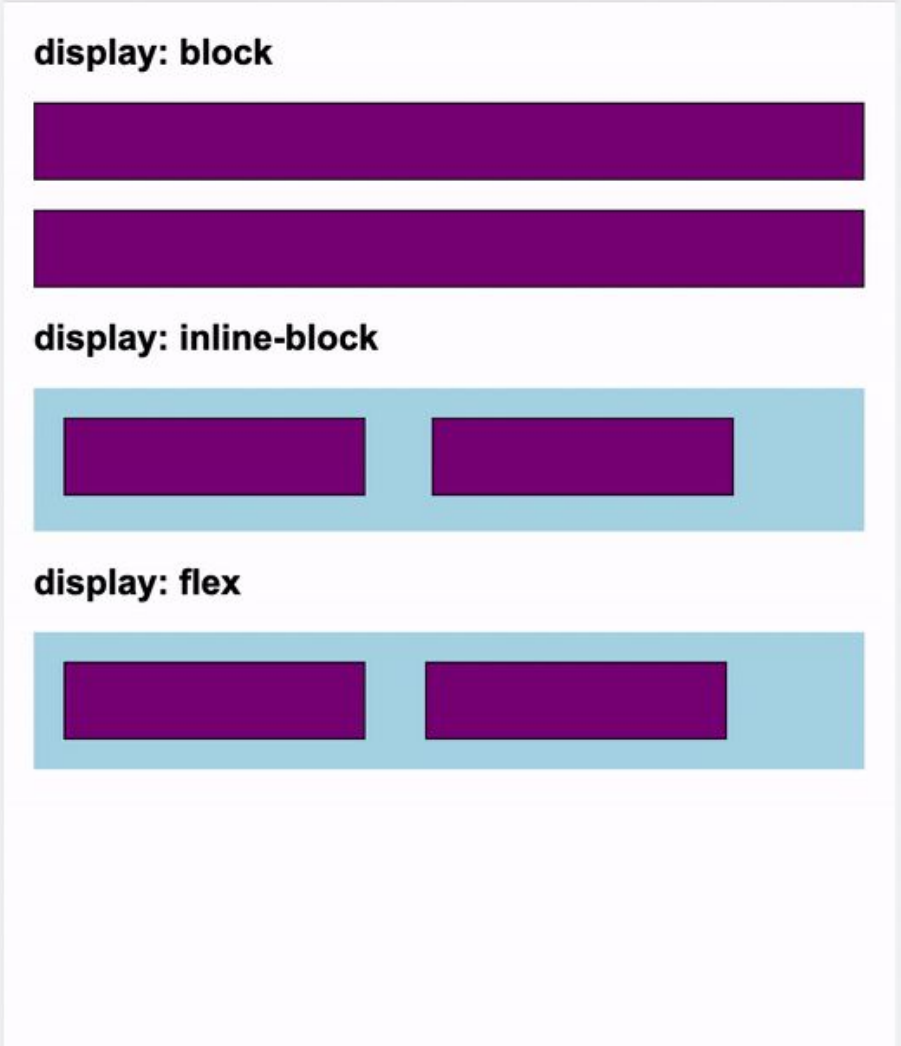
МШП

Теория

Сетки

МШП

Может ли flexbox делать 2 строки, как inline-block?

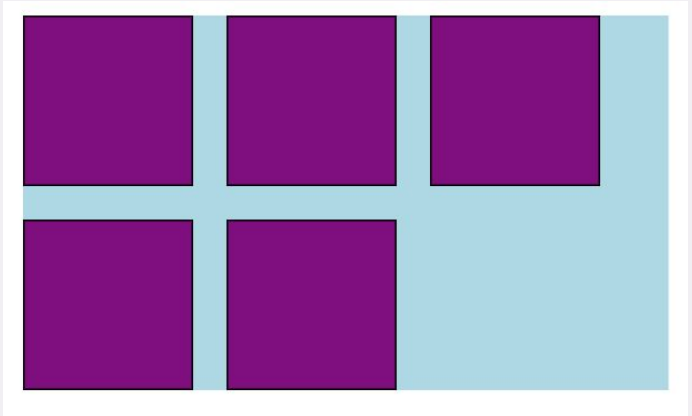


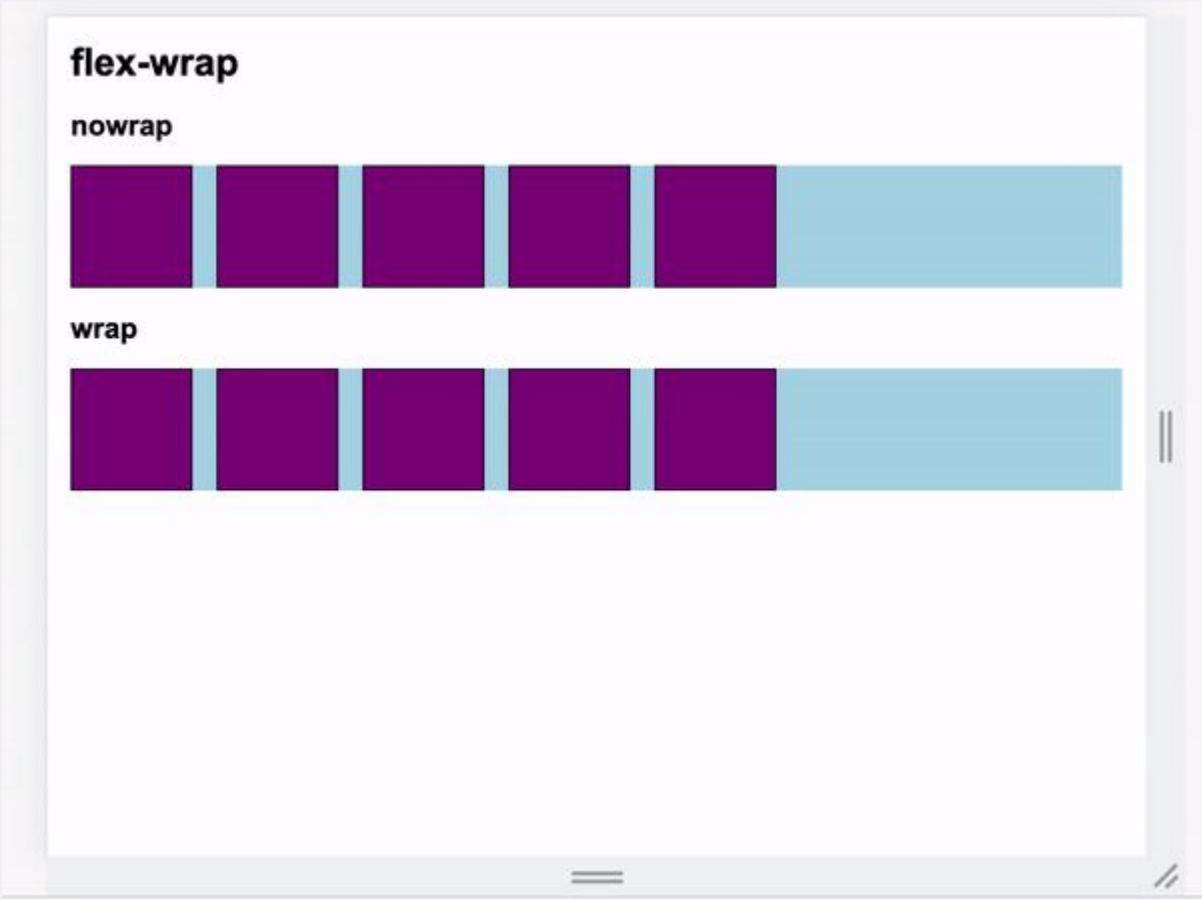
МШП

flex-wrap

перенос элементов

```
.container {
    display: flex;
    gap: 20px;
    flex-wrap: wrap;
}
```




flex-wrap
nowrap
wrap

МШП

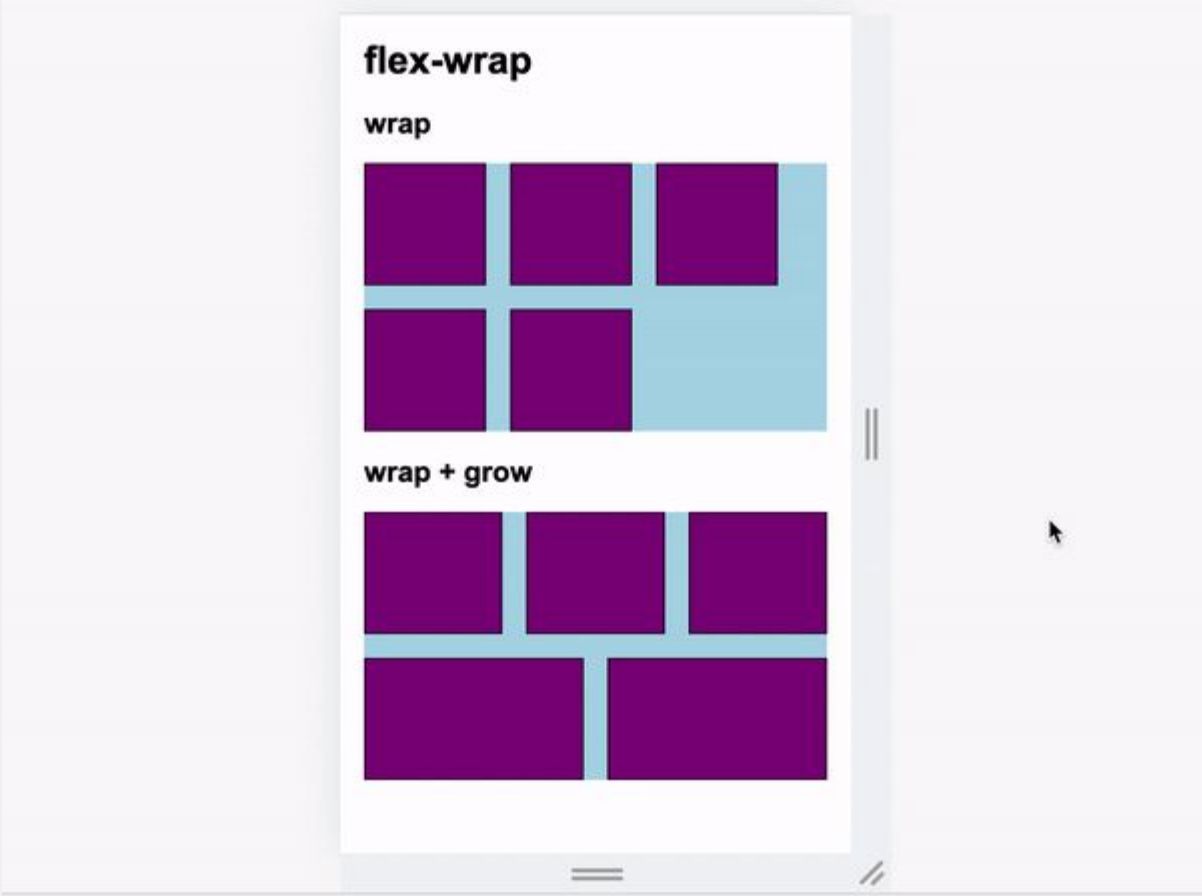
flex-wrap

перенос элементов

```
.container {
    display: flex;
    gap: 20px;
    flex-wrap: wrap;
}

.item {
    flex-grow: 1;
    flex-basis: 200px;
}
```

Ультра комбо для гибкой сетки


flex-wrap
wrap
wrap + grow

МШП

Практика

Сетки

МШП

Цели урока:

- ✓ Познакомились с моделью flexbox
- ✓ Научились создавать гибкие элементы
- ✓ Создали простую сетку