

Управление потоком документа

Свойство position

Позволяет задать положение HTML-элемента относительно текущей позиции, родительского элемента или окна браузера.

`position: static` *default*

`position: relative`

`position: absolute`

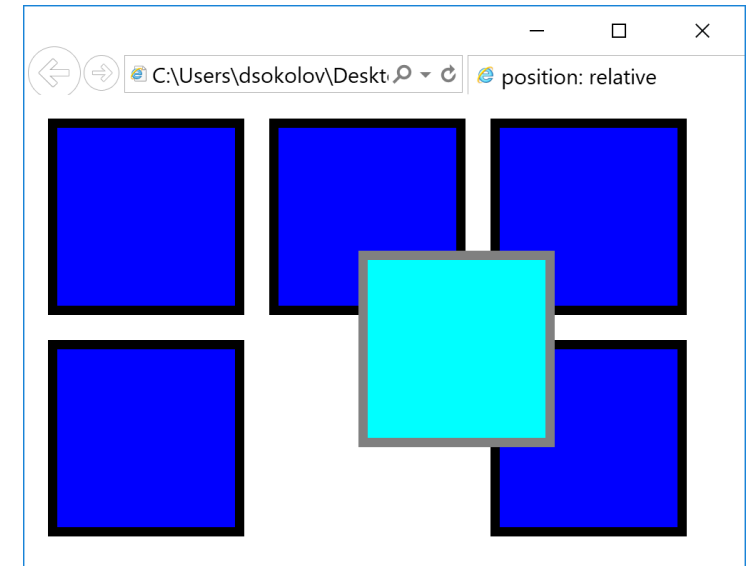
`position: fixed`

Относительное позиционирование

```
<div class="foo"></div>
<div class="foo"></div>
<div class="foo"></div>
<div class="foo"></div>
<div class="foo bar"></div>
<div class="foo"></div>
```

```
.foo {
  display: inline-block;
  width: 100px;
  height: 100px;
  border: 5px solid black;
  margin: 5px;
  background: blue;
}

.bar {
  background: aqua;
  border-color: gray;
  position: relative;
  bottom: 50px;
  left: 50px;
}
```



Абсолютное позиционирование

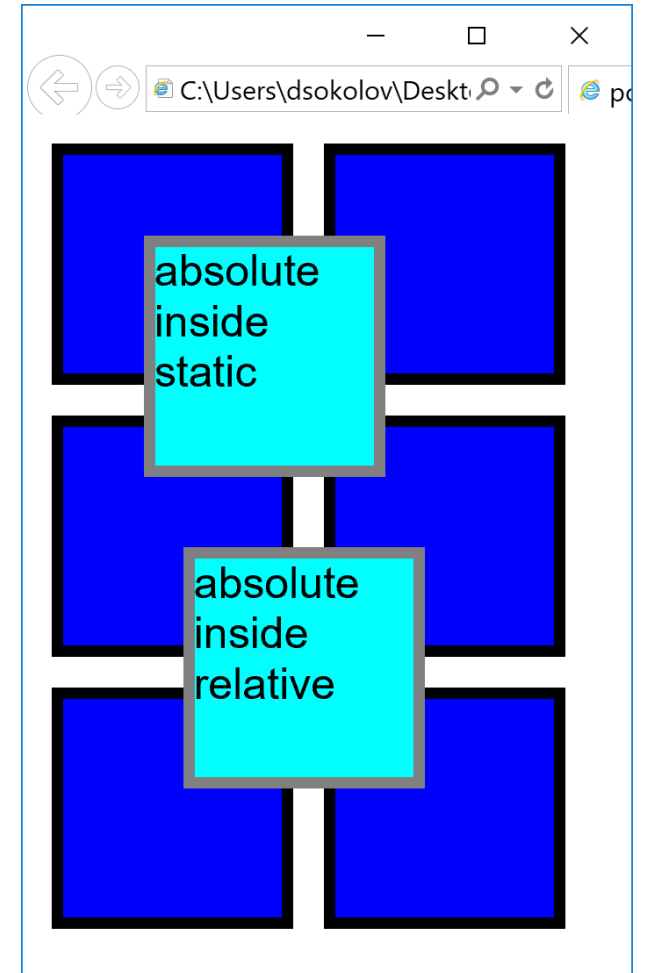
```
<div class="foo"></div>
<div class="foo"></div>
<div class="foo"
  style="position: relative;">
  <div class="bar">
    absolute inside relative
  </div>
</div>
<div class="foo"></div>
<div class="foo">
  <div class="bar">
    absolute inside static
  </div>
</div>
<div class="foo"></div>
```

```
.foo, .bar {
  display: inline-block;
  width: 100px;
  height: 100px;
  margin: 5px;
  border: 5px solid;
}

.foo {
  border-color: black;
  background: blue;
}

.bar {
  border-color: gray;
  background: aqua;
  font: 20px Arial;
}

.bar {
  position: absolute;
  top: 50px;
  left: 50px;
}
```

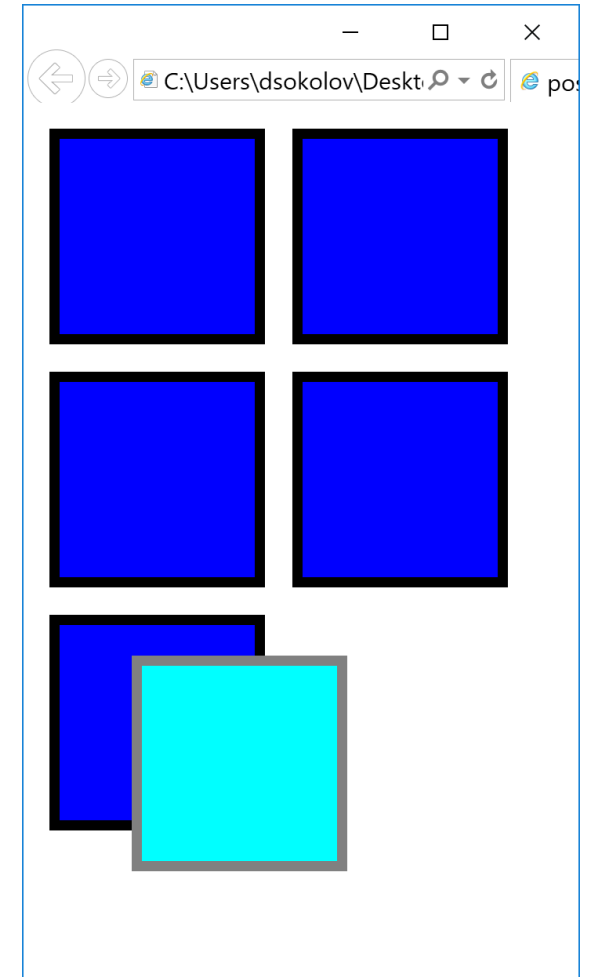


Фиксированное позиционирование

```
<div class="foo"></div>
<div class="foo"></div>
<div class="foo"></div>
<div class="foo"></div>
<div class="foo bar"></div>
<div class="foo"></div>
```

```
.foo {
  display: inline-block;
  width: 100px;
  height: 100px;
  border: 5px solid black;
  margin: 5px;
  background: blue;
}

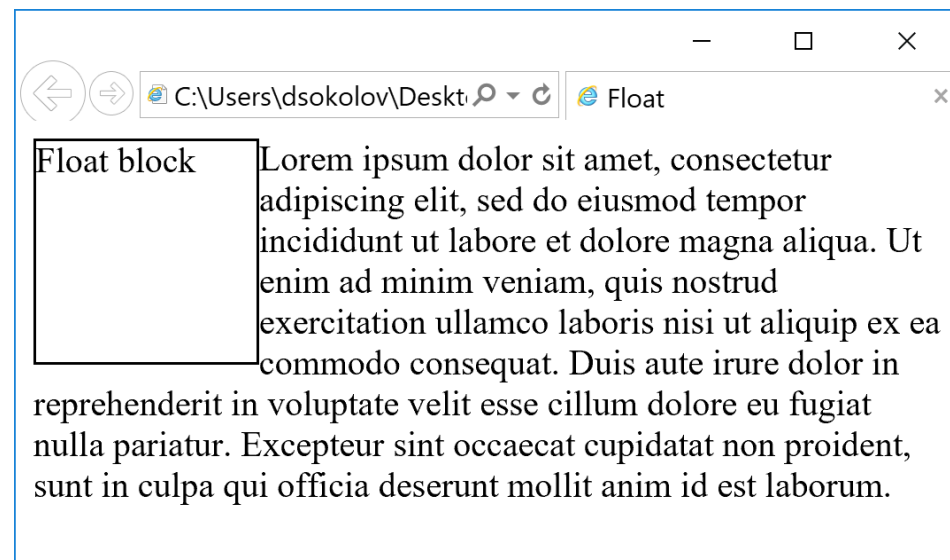
.bar {
  background: aqua;
  border-color: gray;
  position: fixed;
  bottom: 50px;
  left: 50px;
}
```



Плавающие элементы

```
<div>  
  <div class="foo">  
    Float block  
  </div>  
  Lorem ipsum dolor sit amet,  
  consectetur adipiscing elit,  
  sed do eiusmod tempor  
  incididunt ut labore et dolore  
  magna aliqua. Ut enim ad minim  
  veniam, quis nostrud  
  exercitation ullamco laboris  
  nisi ut aliquip ex ea commodo  
  consequat. Duis aute irure  
  dolor in reprehenderit in  
  voluptate velit esse cillum  
  dolore eu fugiat nulla  
  pariatur. Excepteur sint  
  occaecat cupidatat non  
  proident, sunt in culpa qui  
  officia deserunt mollit anim id  
  est laborum.  
</div>
```

```
.foo {  
  width: 100px;  
  height: 100px;  
  border: 1px solid black;  
  float: left;  
}
```

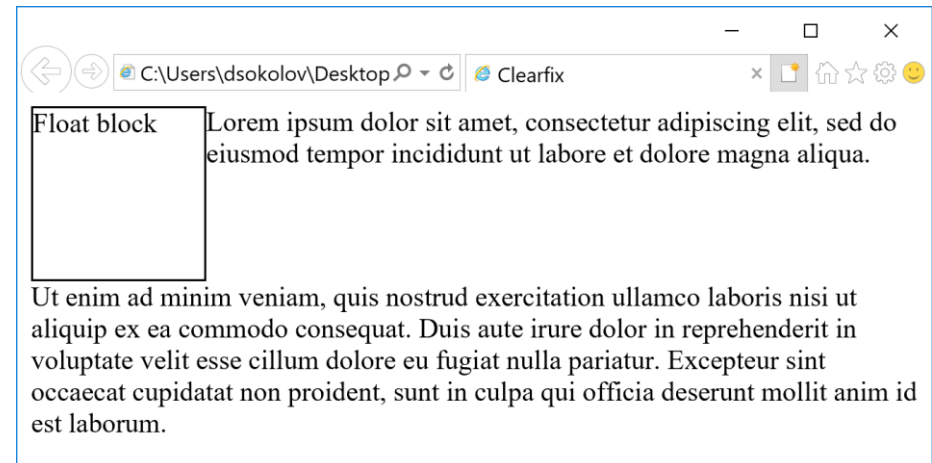


Отмена обтекания

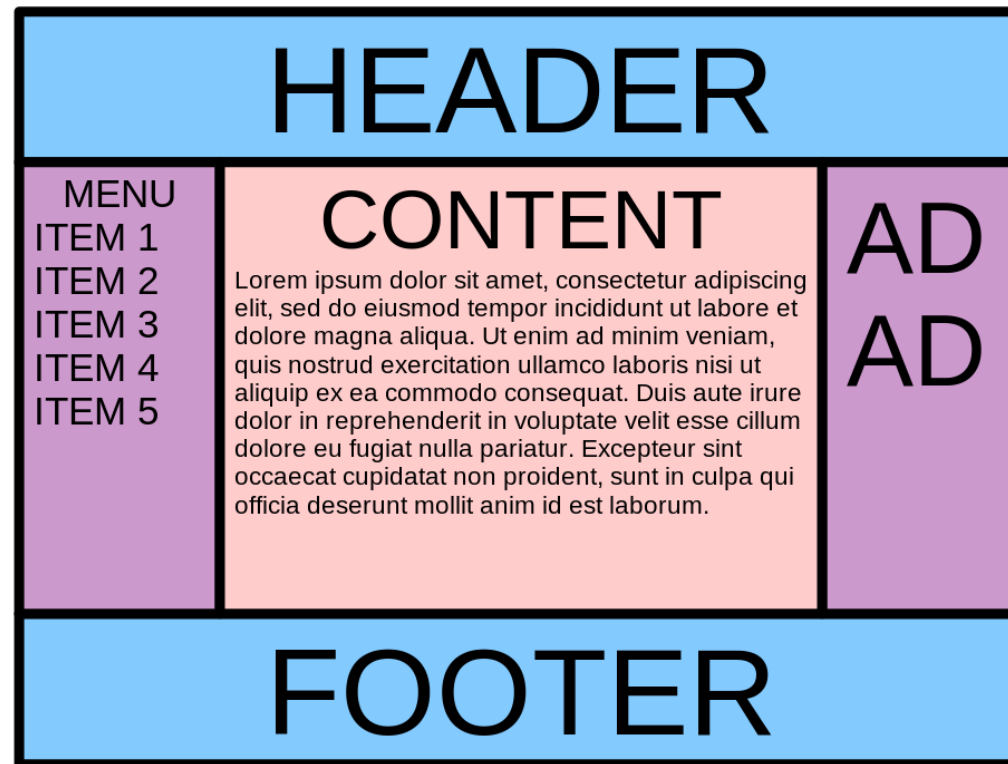
```
<div>
  <div class="foo">
    Float block
  </div>
  Lorem ipsum dolor sit amet,
  consectetur adipiscing elit, sed
  do eiusmod tempor incididunt ut
  labore et dolore magna aliqua.
</div>
<div class="clearfix">
  Ut enim ad minim veniam, quis
  nostrud exercitation ullamco
  laboris nisi ut aliquip ex ea
  commodo consequat. Duis aute
  irure dolor in reprehenderit in
  voluptate velit esse cillum
  dolore eu fugiat nulla pariatur.
  Excepteur sint occaecat
  cupidatat non proident, sunt in
  culpa qui officia deserunt
  mollit anim id est laborum.
</div>
```

```
.foo {
  width: 100px;
  height: 100px;
  border: 1px solid black;
  float: left;
}

.clearfix:after {
  content: "";
  clear: both;
  display: block;
}
```



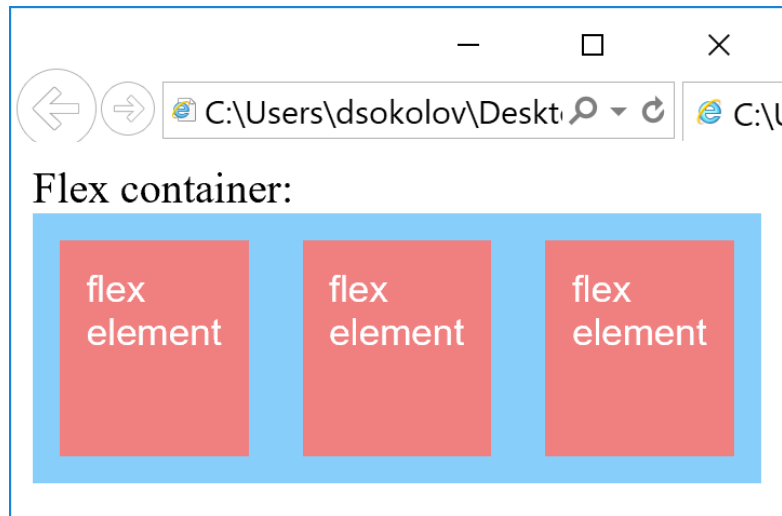
Holy Grail Layout



Flexbox

Flex container:

```
<div class="container">
  <div class="element">flex element</div>
  <div class="element">flex element</div>
  <div class="element">flex element</div>
</div>
```

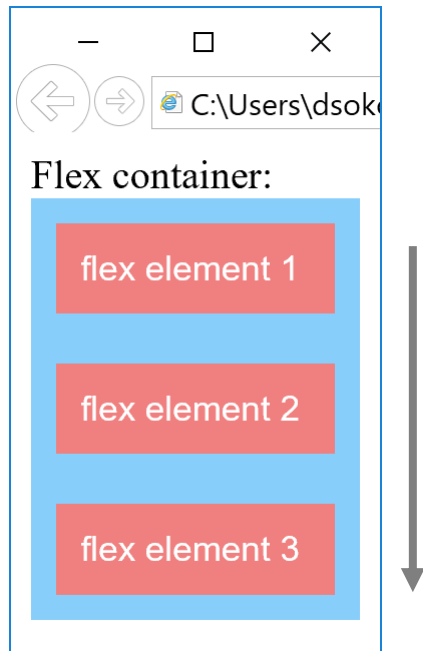


```
.container {
  display: flex;
  height: 100px;
  background: lightskyblue;
}
```

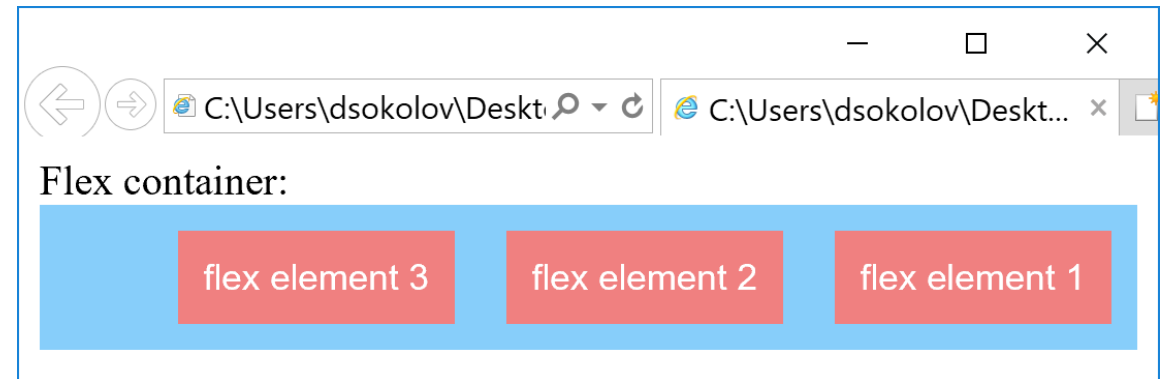
```
.element {
  width: 50px;
  background: lightcoral;
  margin: 10px;
  padding: 10px;
  font: 14px Arial;
  color: white;
}
```

Направление главной оси

```
.container {  
  display: flex;  
  flex-direction: column;  
}
```

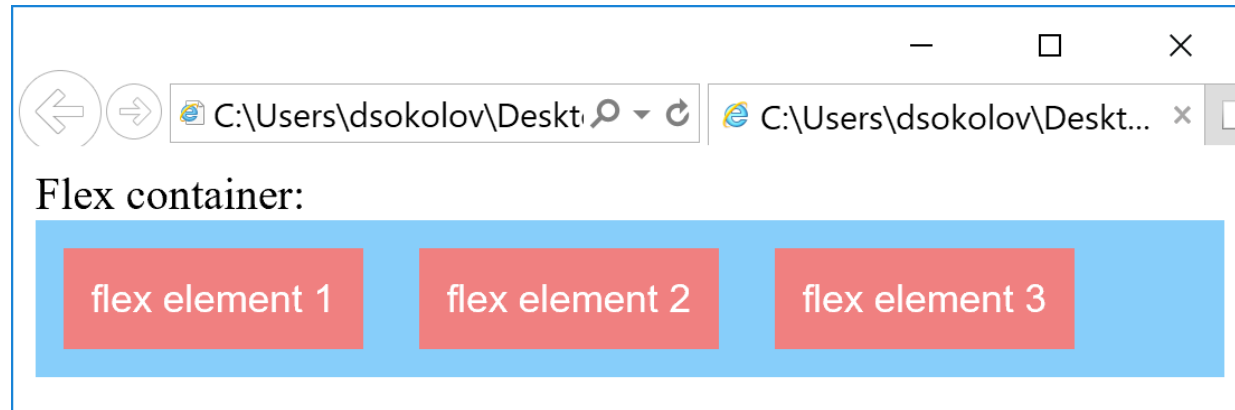


```
.container {  
  display: flex;  
  flex-direction: row-reverse;  
}
```



```
flex-direction: row;  
flex-direction: row-reverse;  
flex-direction: column;  
flex-direction: column-reverse;
```

Направление поперечной оси



Поперечная ось (Cross Axis)

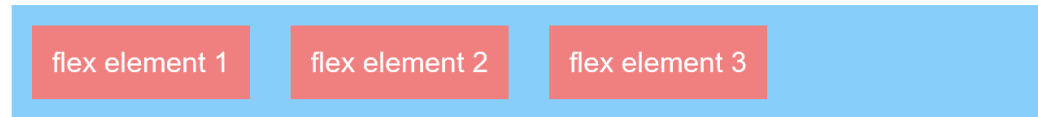


Главная ось (Main Axis)

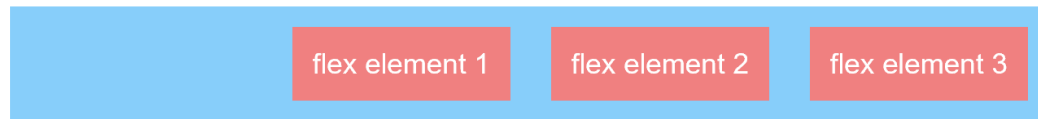
Поперечная ось всегда перпендикулярна главной оси и поворачивается вместе с ней:

- Если главная ось направлена горизонтально, то поперечная ось смотрит вниз.
- Если главная ось направлена вертикально, то поперечная ось смотрит направо.

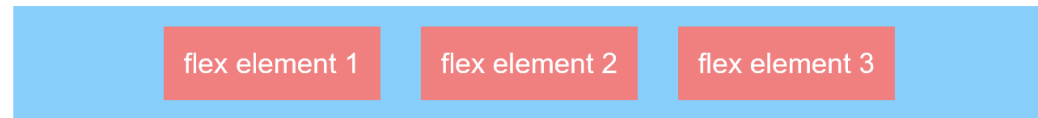
Распределение элементов вдоль главной оси



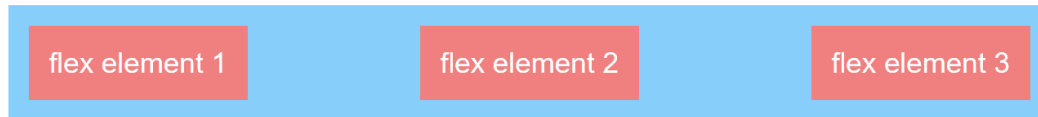
`justify-content: flex-start;`



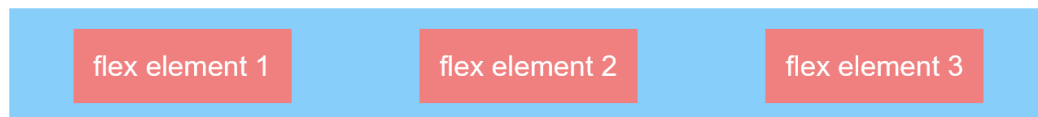
`justify-content: flex-end;`



`justify-content: center;`

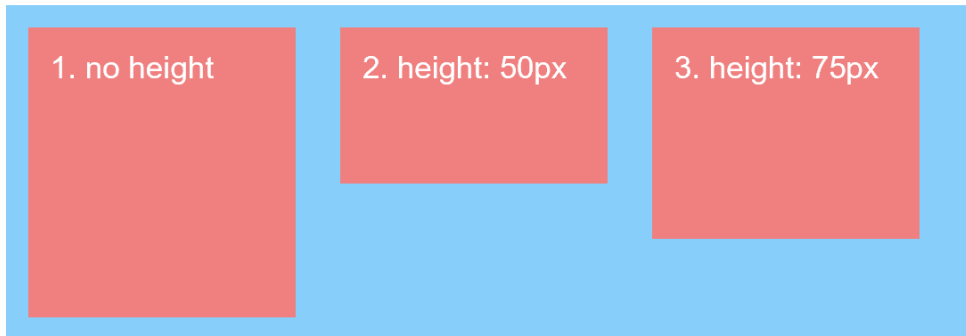


`justify-content: space-between;`



`justify-content: space-around;`

Распределение элементов вдоль поперечной оси



`align-items: flex-start;` (default)



`align-items: flex-end;` (flex-start)



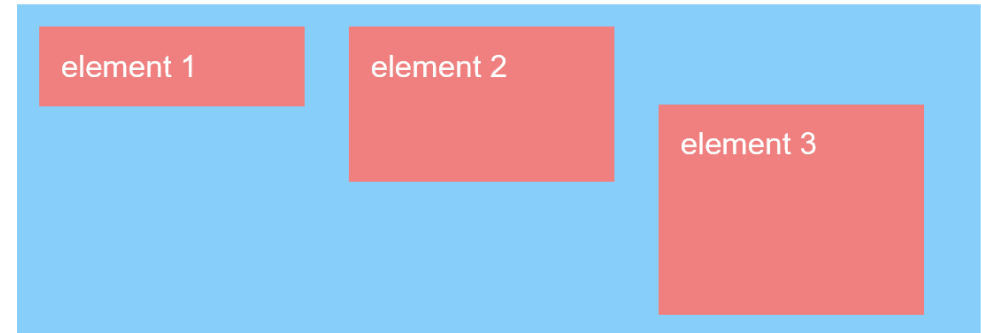
`align-items: center;`



`align-items: baseline;`

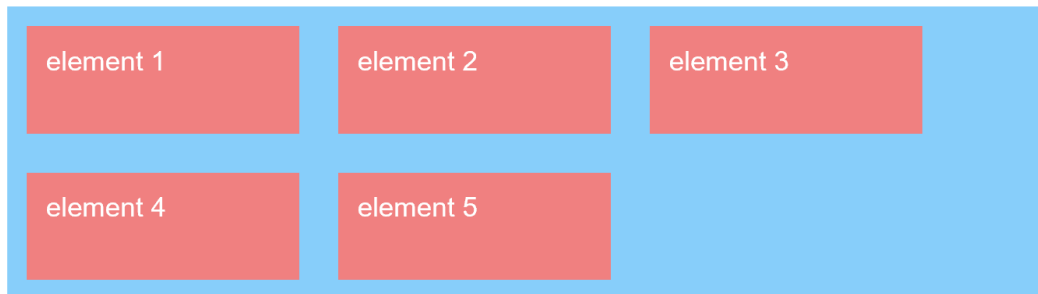
Эгоистичное выравнивание вдоль поперечной оси

```
.container {  
  display: flex;  
  align-items: flex-start;  
}  
  
.element-3 {  
  align-self: flex-end;  
}
```

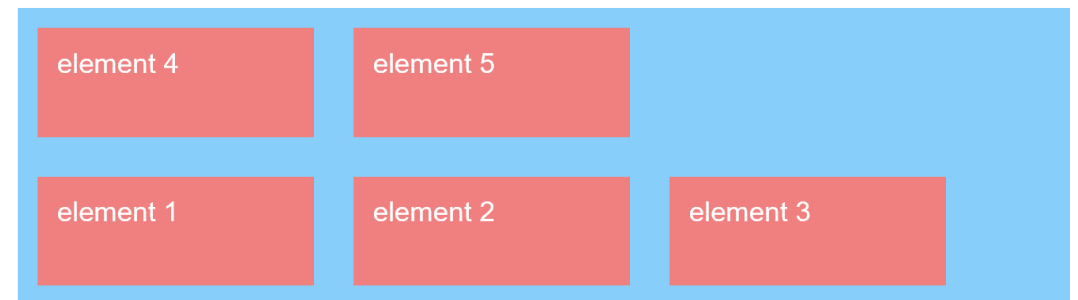


Перенос флекс-элементов на новую строку

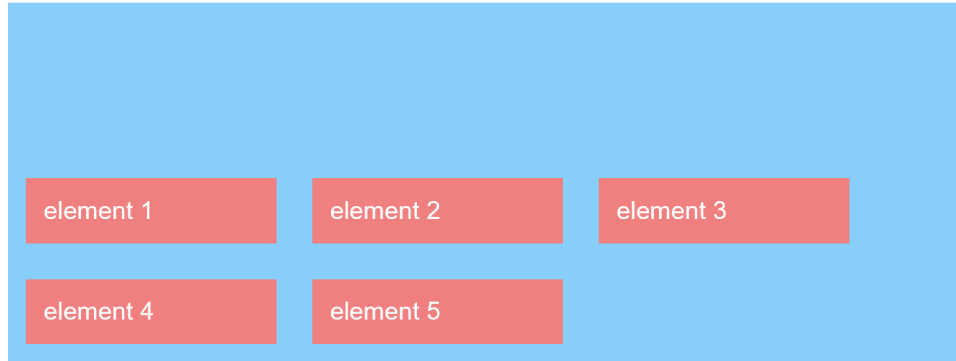
```
.container {  
  display: flex;  
  flex-wrap: wrap;  
}
```



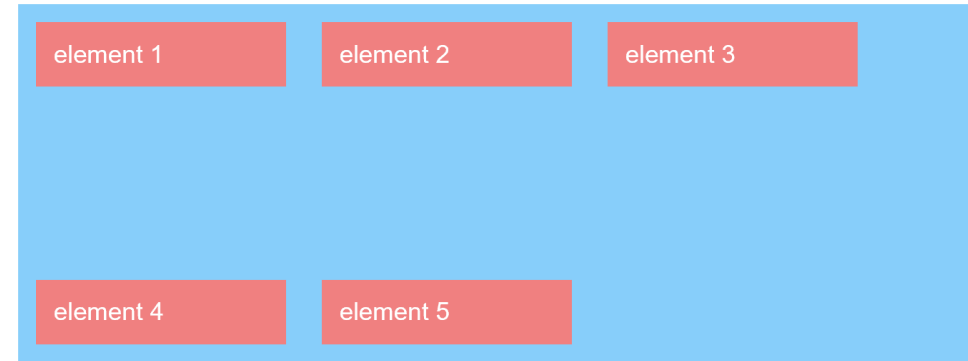
```
.container {  
  display: flex;  
  flex-wrap: wrap-reverse;  
}
```



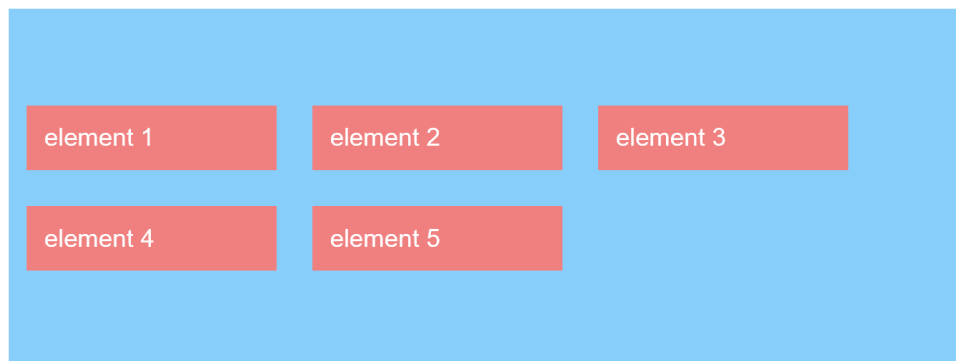
Распределение строк флекс-контейнера



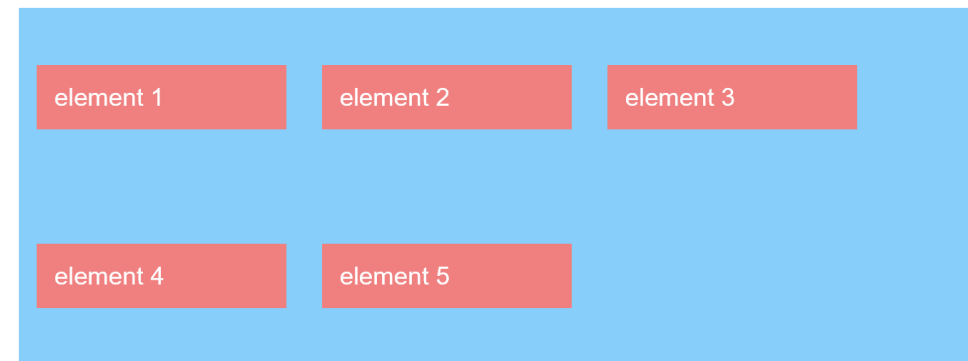
`align-content: flex-end;`



`align-content: space-between;`



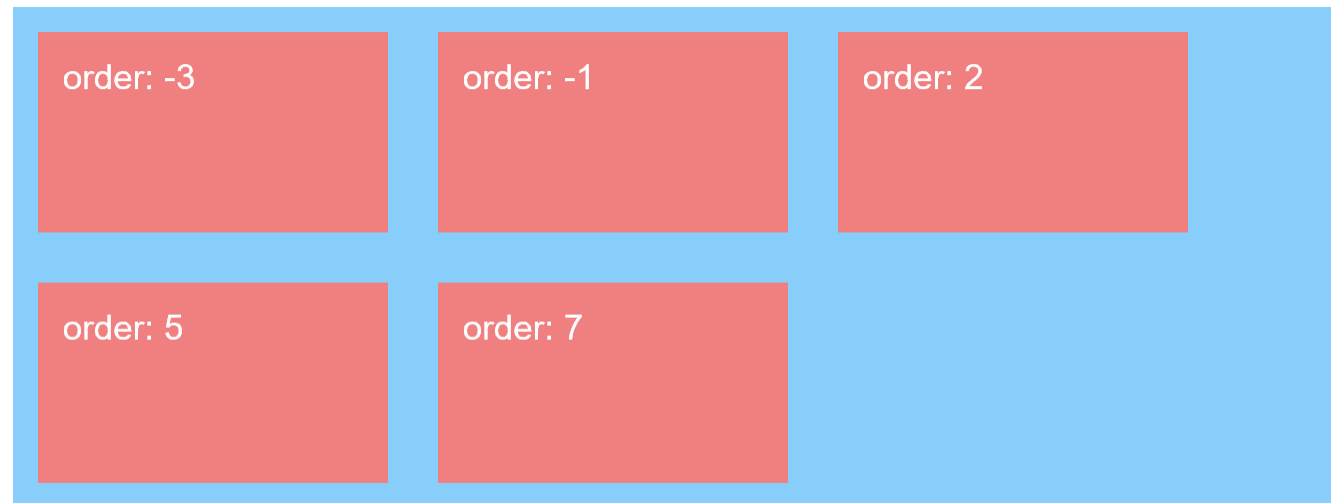
`align-content: center;`



`align-content: space-around;`

Порядковый номер флекс-элемента

```
<div class="container">  
  <div class="item" style="order: 5">order: 5</div>  
  <div class="item" style="order: -3">order: -3</div>  
  <div class="item" style="order: 2">order: 2</div>  
  <div class="item" style="order: -1">order: -1</div>  
  <div class="item" style="order: 7">order: 7</div>  
</div>
```



Коэффициент растягивания элементов

```
<div class="container">  
  <div class="item" style="flex-grow: 1">flex-grow: 1</div>  
  <div class="item" style="flex-grow: 2">flex-grow: 2</div>  
  <div class="item" style="flex-grow: 3">flex-grow: 3</div>  
</div>
```



Коэффициент сжатия элементов

```
<div class="container">  
  <div class="item" style="flex-basis: 400px; flex-shrink: 1">flex-shrink: 1</div>  
  <div class="item" style="flex-basis: 400px; flex-shrink: 2">flex-shrink: 2</div>  
  <div class="item" style="flex-basis: 400px; flex-shrink: 3">flex-shrink: 3</div>  
</div>
```

flex-shrink: 1

flex-shrink: 2

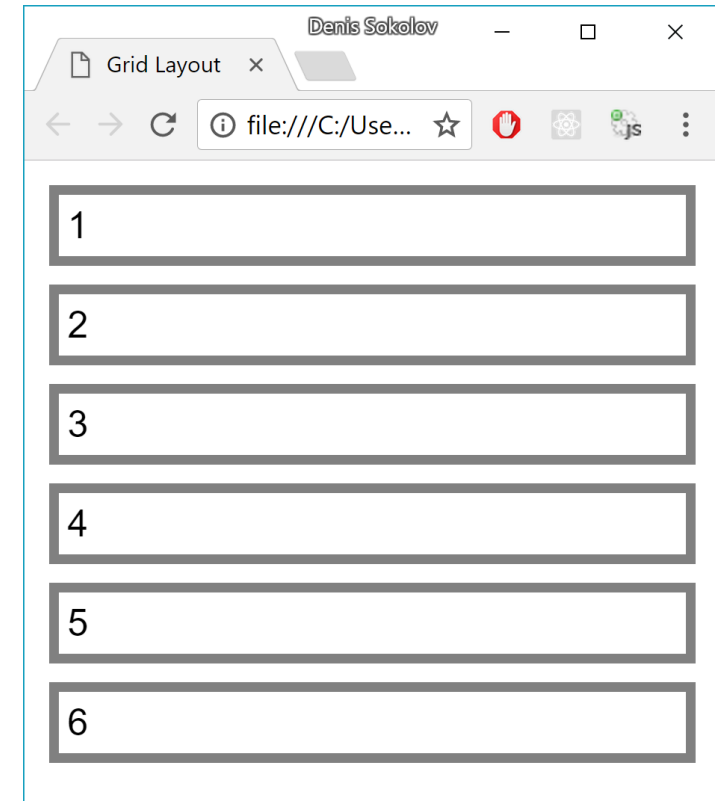
flex-shrink: 3

Grid Layout

```
<div class="container">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item">4</div>
  <div class="item">5</div>
  <div class="item">6</div>
</div>
```

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

```
.item {
  font: 20px Arial;
  border: 5px solid gray;
  margin: 5px;
  padding: 5px;
}
```



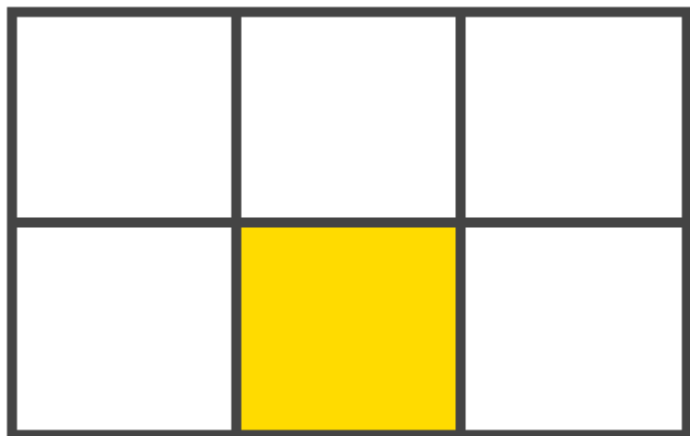
Терминология



Grid line (column)



Grid Track



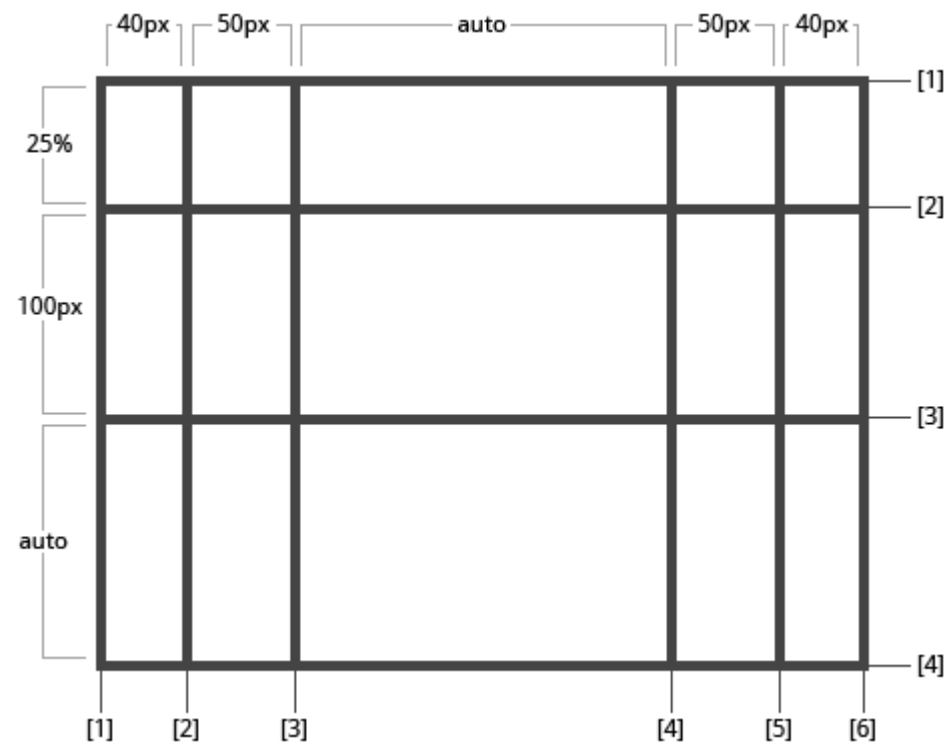
Grid Cell



Grid Area

Размеры строк и колонок

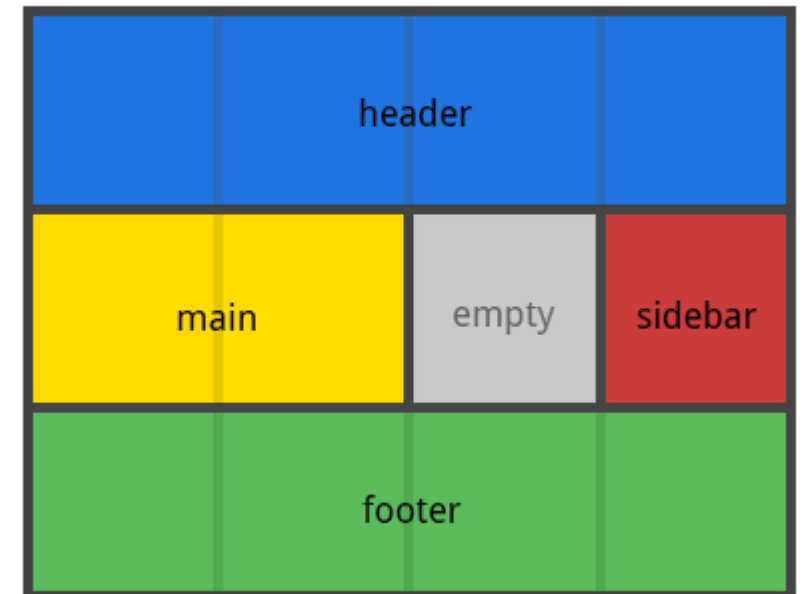
```
.container {  
  display: grid;  
  grid-template-columns:  
    40px 50px auto 50px 40px;  
  grid-template-rows:  
    25% 100px auto;  
}
```



Формирование областей по линиям

```
.header {  
  grid-column-start: 1;  
  grid-column-end: 5;  
  grid-row-start: 1;  
  grid-row-end: 2;  
  background: blue;  
}  
  
.footer {  
  grid-column-start: 1;  
  grid-column-end: span 4;  
  grid-row-start: 3;  
  background: green;  
}
```

```
.main {  
  grid-column: 1 / 3;  
  grid-row: 2 / 3;  
  background: yellow;  
}  
  
.sidebar {  
  grid-area: 2 / 4 / 3 / 5;  
  background: red;  
}  
  
<div class="container">  
  <div class="header"></div>  
  <div class="footer"></div>  
  <div class="main"></div>  
  <div class="sidebar"></div>  
</div>
```



Формирование областей по шаблонам

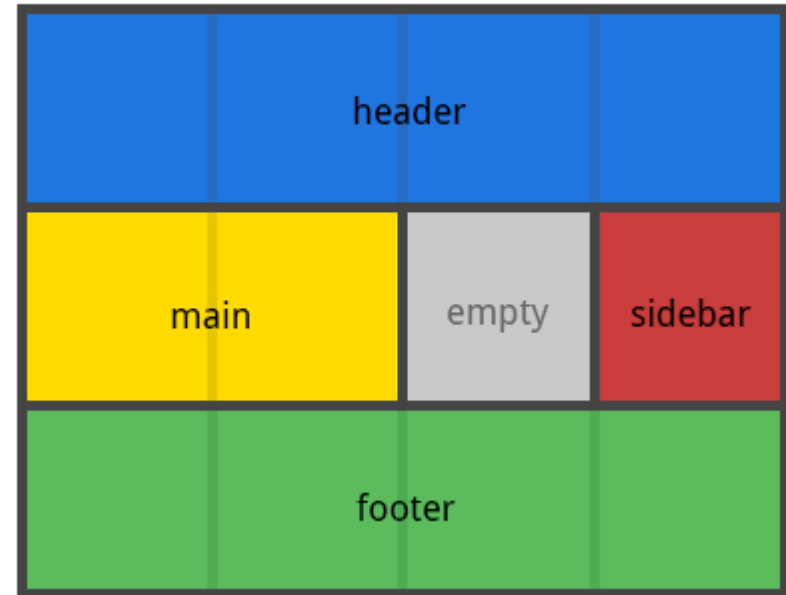
```
.container {  
  display: grid;  
  grid-template-columns: 50px 50px 50px 50px;  
  grid-template-rows: auto;  
  grid-template-areas:  
    "header header header header"  
    "main main . sidebar"  
    "footer footer footer footer";  
}
```

```
.item-a { grid-area: header; }
```

```
.item-b { grid-area: main; }
```

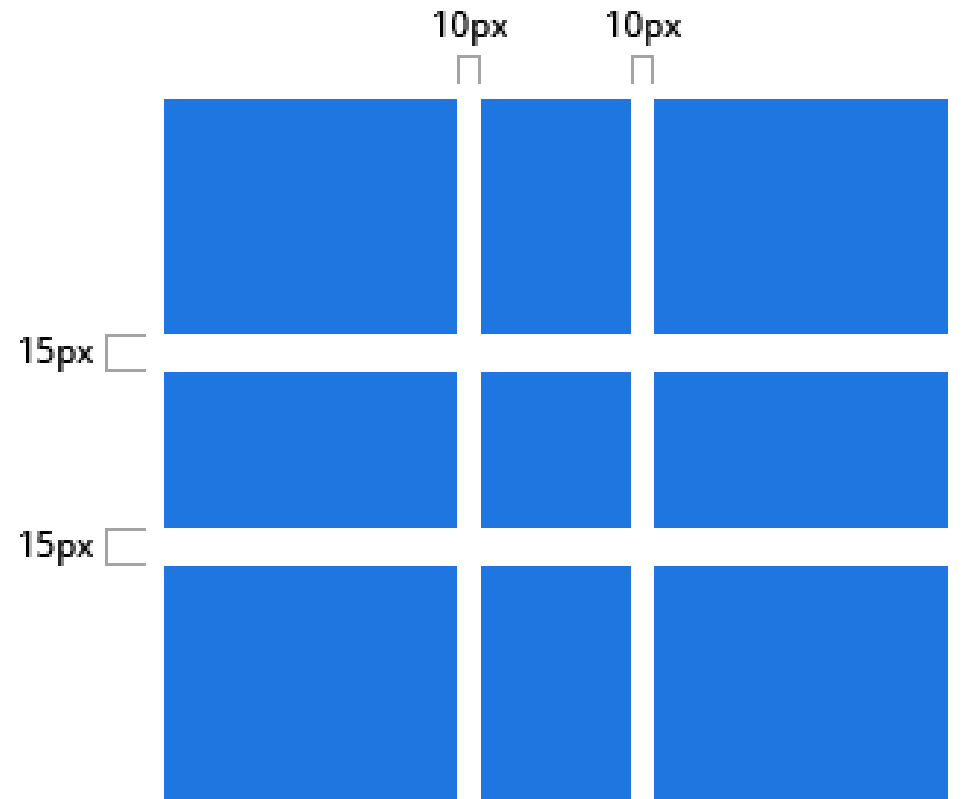
```
.item-c { grid-area: sidebar; }
```

```
.item-d { grid-area: footer; }
```

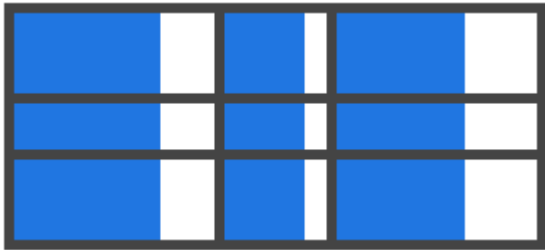


Размеры линий

```
.container {  
  display: grid;  
  grid-template-columns: 100px 50px 100px;  
  grid-template-rows: 80px auto 80px;  
  grid-column-gap: 10px;  
  grid-row-gap: 15px;  
}
```



Горизонтальное выравнивание контента внутри ячейки



`justify-items: start;`



`justify-items: end;`



`justify-items: center;`



`justify-items: stretch;` (default)

`justify-self` – для отдельной ячейки

Вертикальное выравнивание контента внутри ячейки



`align-items: start;`



`align-items: end;`



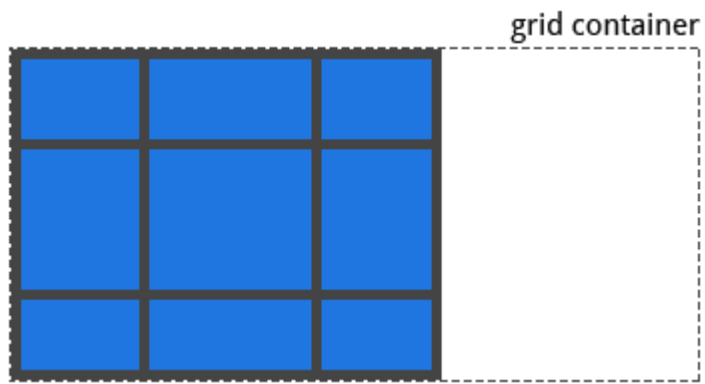
`align-items: center;`



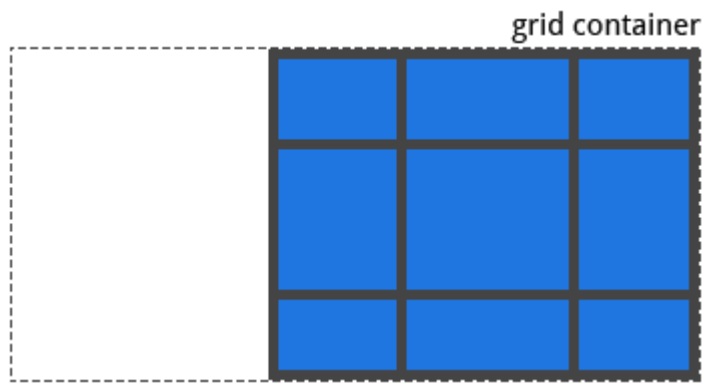
`align-items: stretch;` (default)

`align-self` — для отдельной ячейки

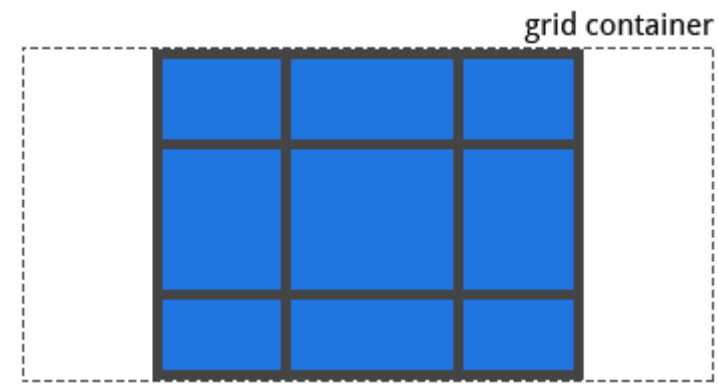
Горизонтальное выравнивание ячеек



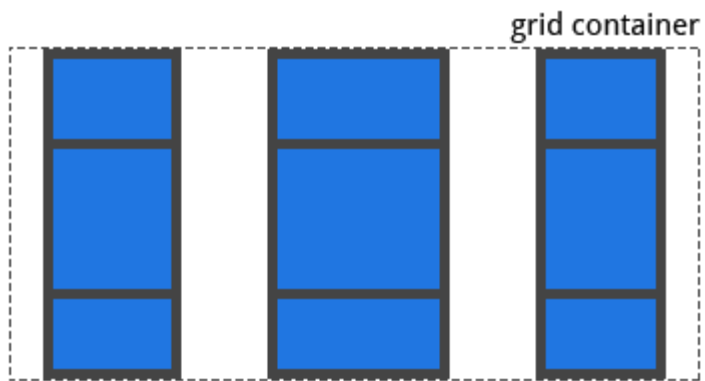
`justify-content: start;`



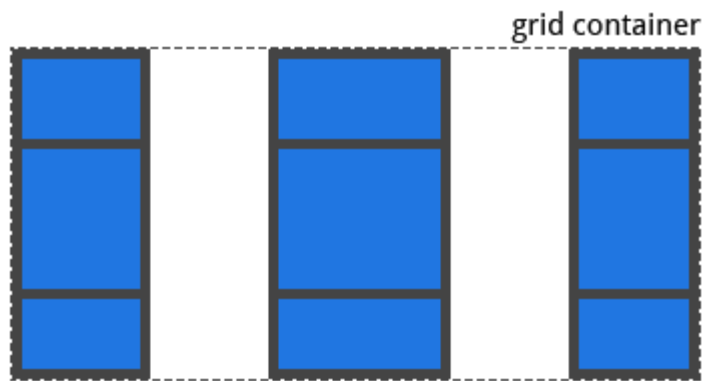
`justify-content: end;`



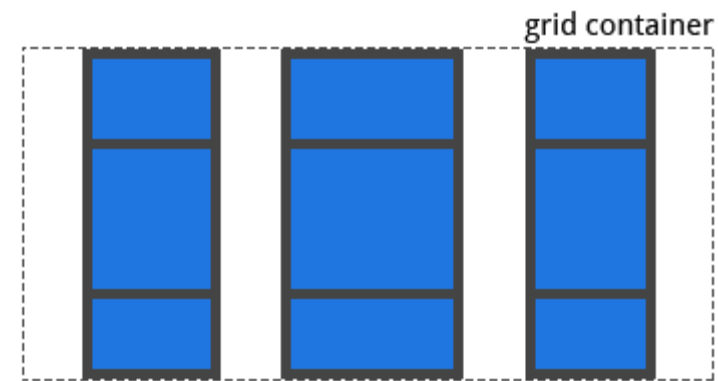
`justify-content: center;`



`justify-content: space-around;`

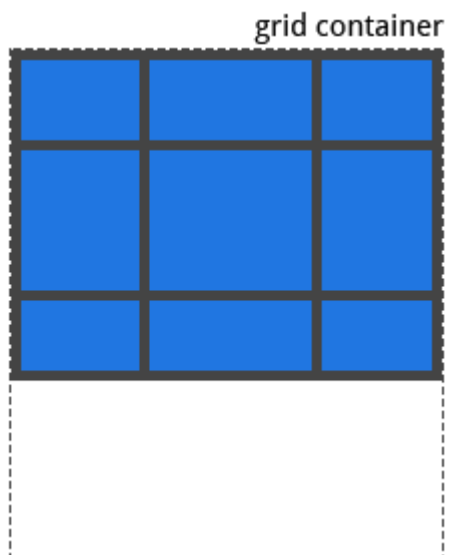


`justify-content: space-between;`



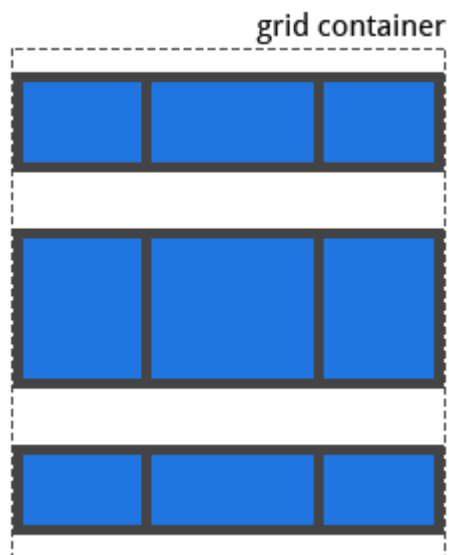
`justify-content: space-evenly;`

Вертикальное выравнивание ячеек

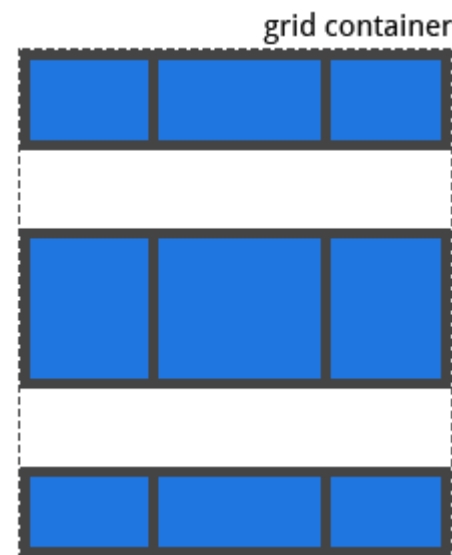


`align-content:`
`start;`

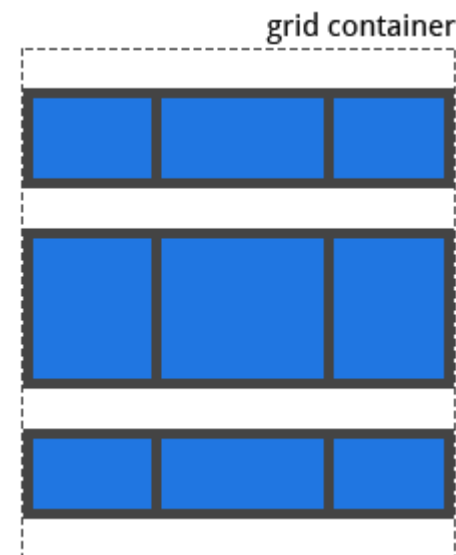
(end, center)



`align-content:`
`space-around;`



`align-content:`
`space-between;`



`align-content:`
`space-evenly;`