Web-программирование

М3403-3406

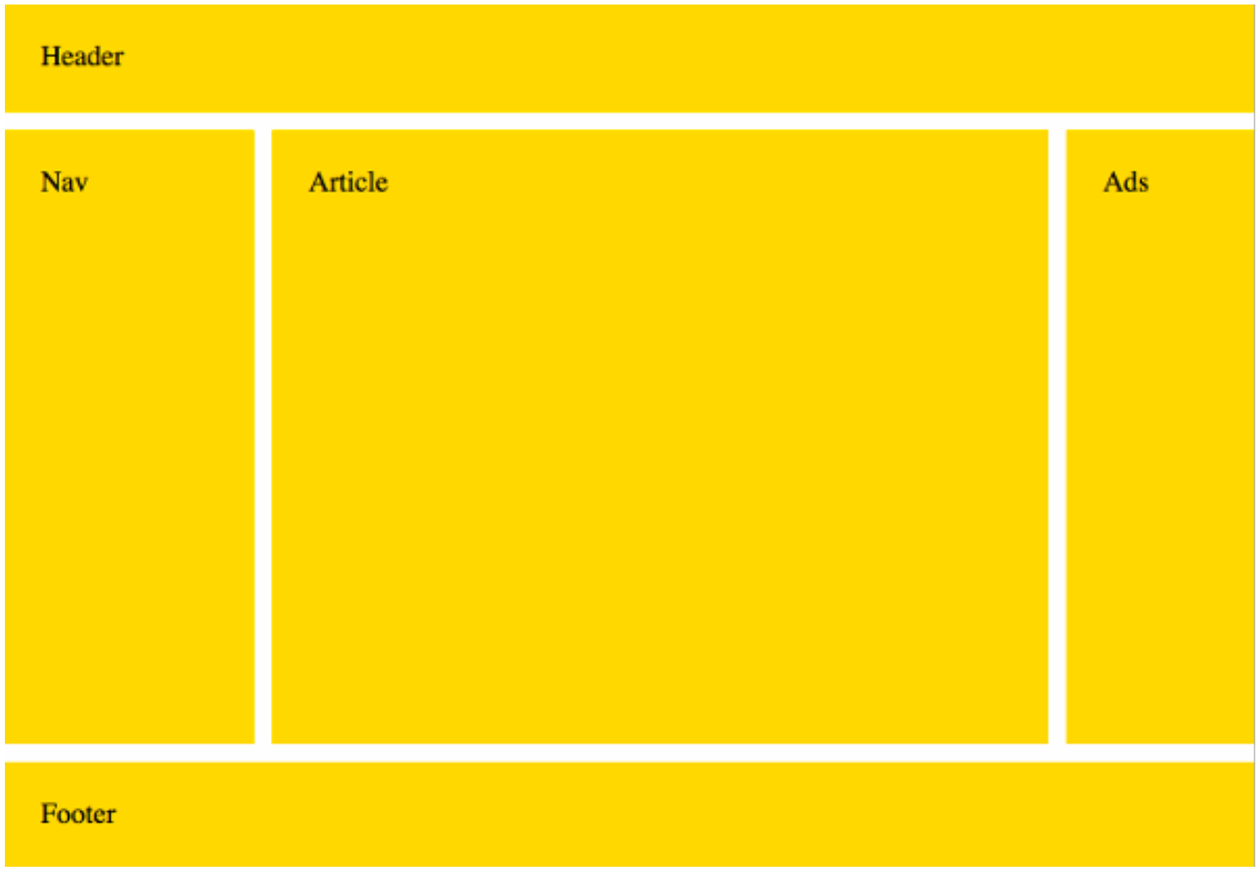
Руслан Владимирович Посевкин

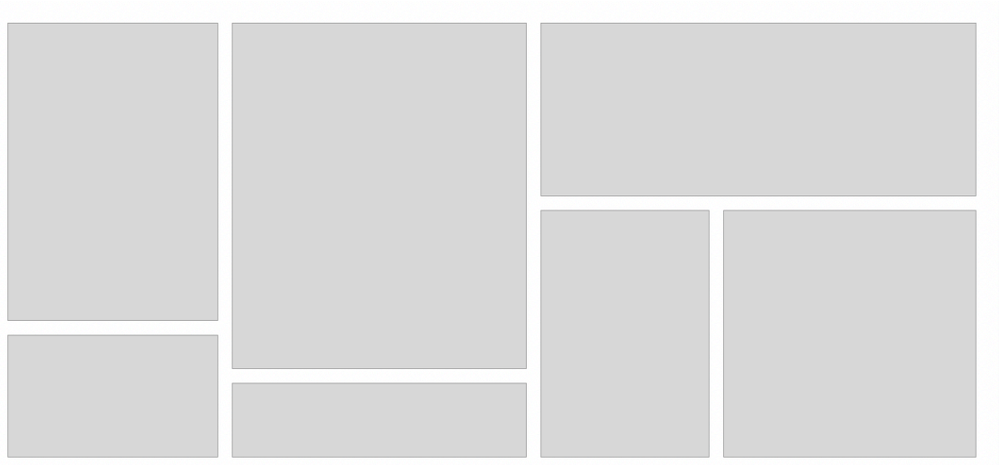
https://vk.com/itmowebdev19

Эволюция сеток в вёрстке

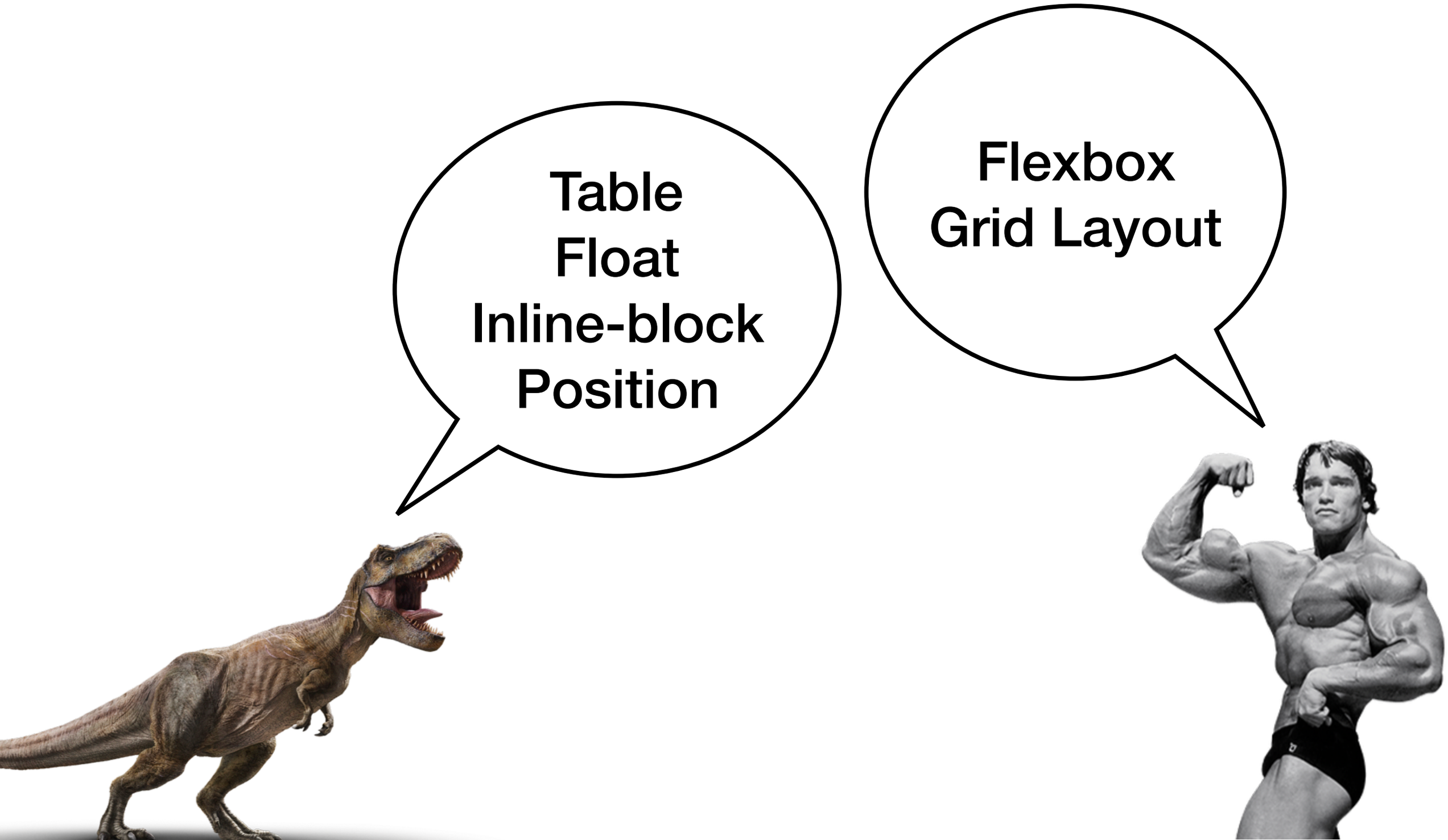
Сетка в журнальной вёрстке

Сетка в вебе





Эволюция подхода



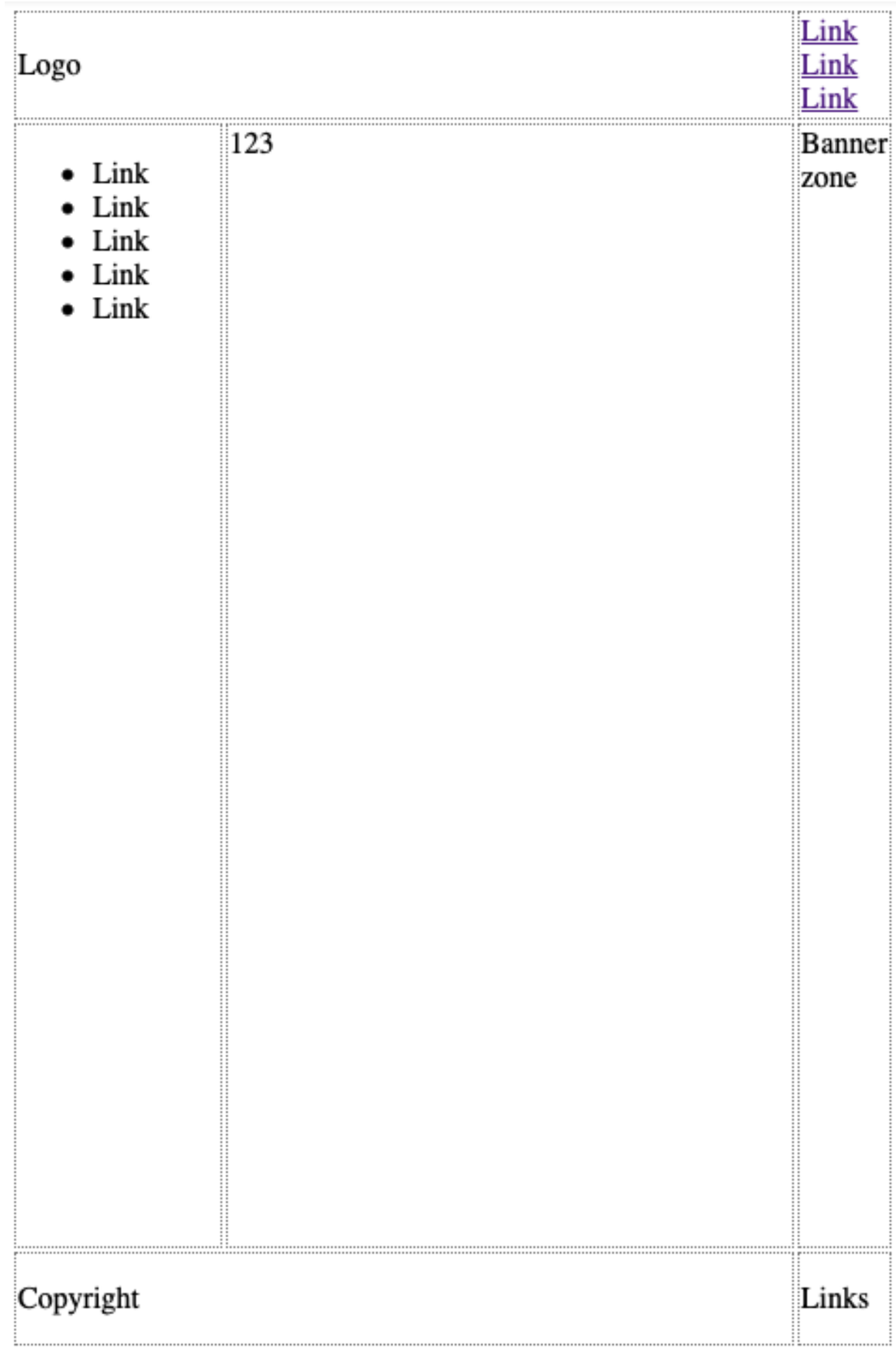
Табличная вёрстка

- `<table>`
- `display: table`
- Для таблиц
- Для email-рассылок
- Не для сетки

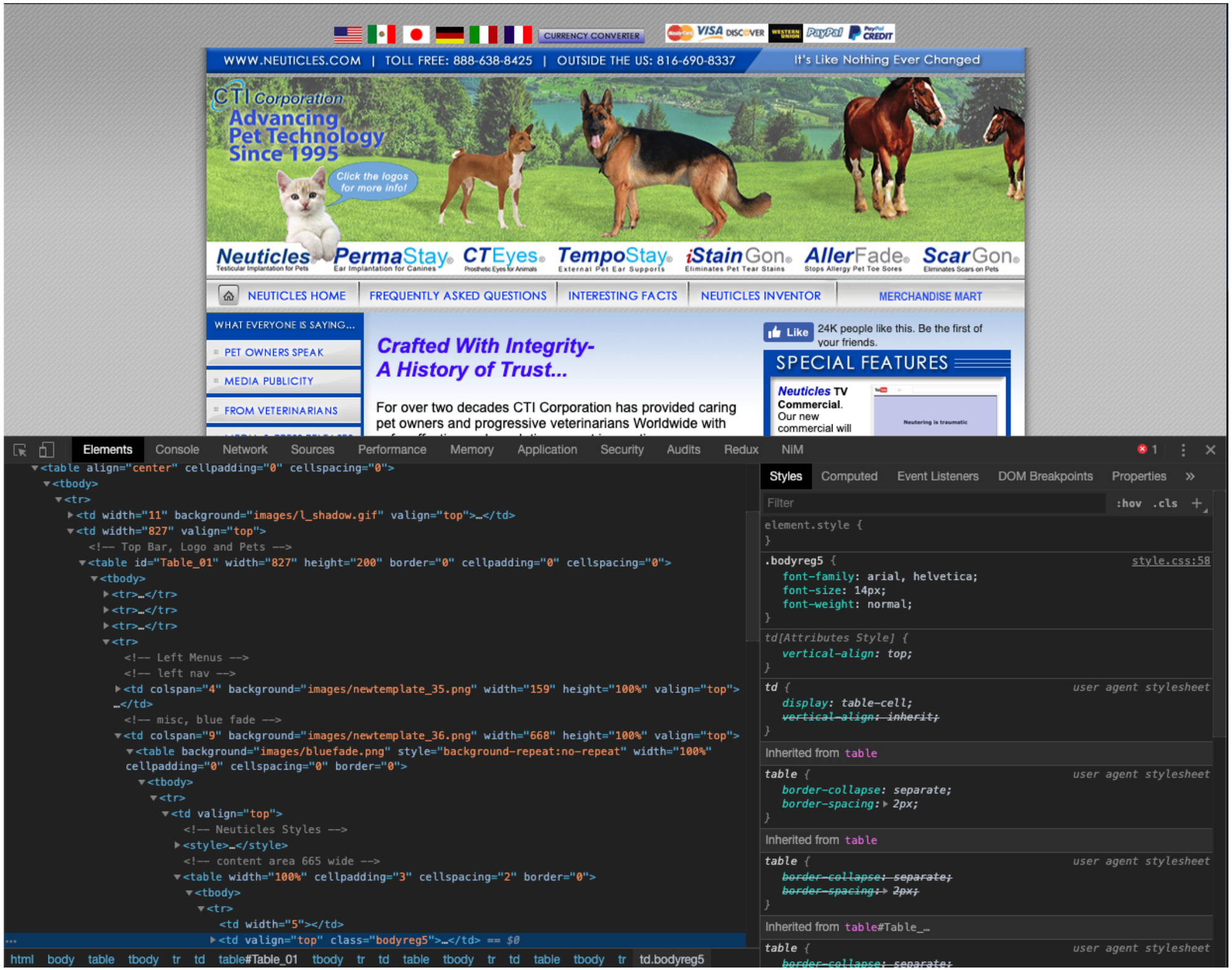
	A	B	C	D
1	**Наименование**	**Количество**	**Цена**	**Сумма**
2	Сыр	213	300	
3	Масло	224	32	
4	Молоко	352	21	
5	Хлеб	120	22	

Табличная вёрстка

```
<table>
    <tbody>
      <tr class='header'>
        <td colspan="2">Logo</td>
        <td>
          <a href="">Link</a>
          <a href="">Link</a>
          <a href="">Link</a>
        </td>
      </tr>
      <tr class="content">
        <td class="menu_vertical" valign="top">
          <ul>
            <li>Link</li>
            <li>Link</li>
            <li>Link</li>
            <li>Link</li>
          </ul>
        </td>
        <td class="text" valign="top">
          123
        </td>
        <td valign="top">Banner zone</td>
      </tr>
      <tr class="footer">
        <td colspan="2">Copyright</td>
        <td>Links</td>
      </tr>
  </tbody>
</table>
```



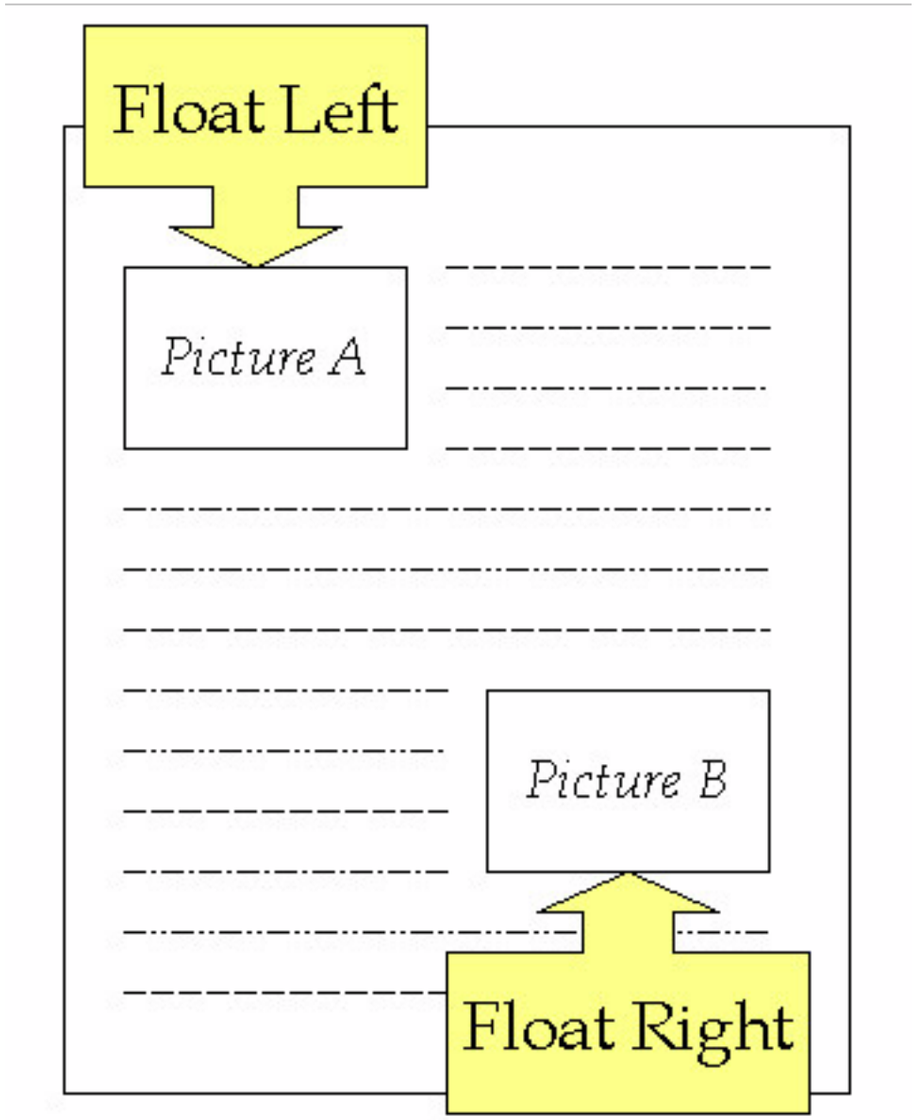
Табличная вёрстка



https://neuticles.com

Float – обтекание элементов

- `float: left | right`
- Для обтекания изображений в тексте
- Выпадает из потока документа
- Нужен clearfix
- Делает элемент блочным



Float: пример

```
<img src="..." />
<p>Lorem ipsum...</p>
```

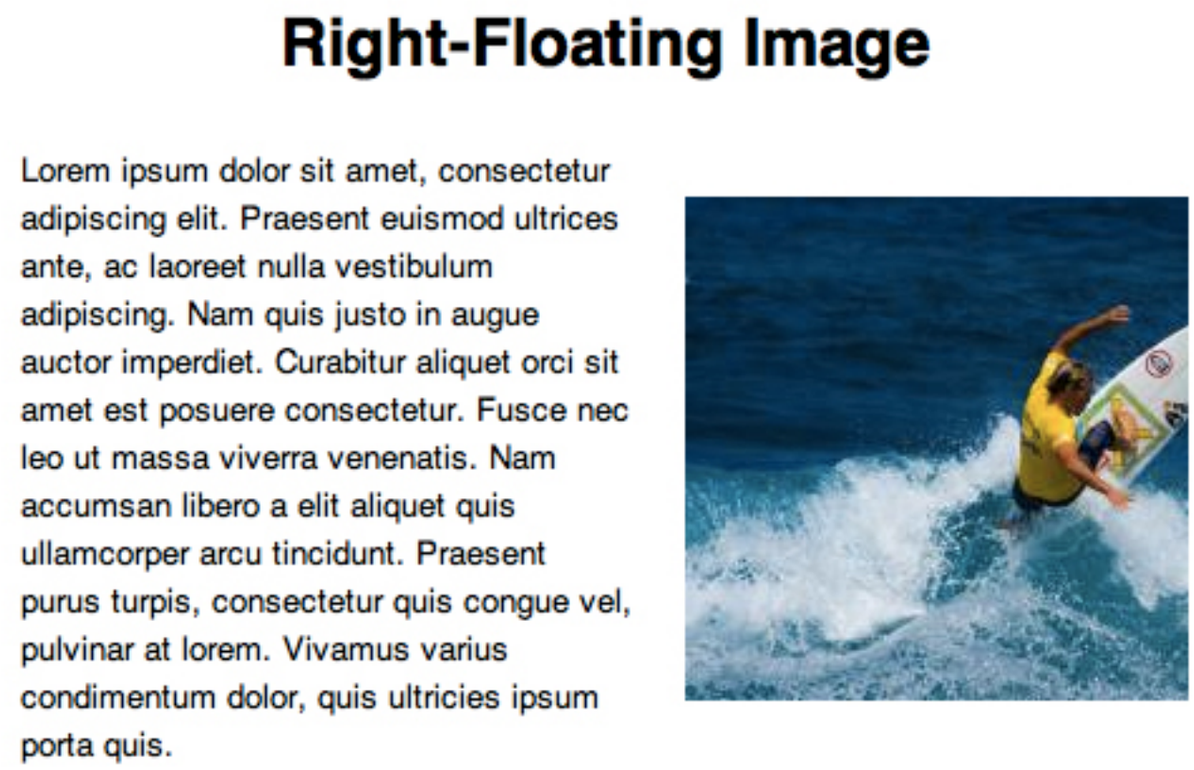
The Normal Document Flow

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent euismod ultrices ante, ac laoreet nulla vestibulum adipiscing. Nam quis justo in augue auctor imperdiet. Curabitur aliquet orci sit amet est posuere consectetur. Fusce nec leo ut massa viverra venenatis. Nam accumsan libero a elit aliquet quis ullamcorper arcu tincidunt. Praesent purus turpis, consectetur quis congue vel, pulvinar at lorem. Vivamus varius condimentum dolor, quis ultricies ipsum porta quis.

Float: пример

Разместим текст и изображение на одной строке

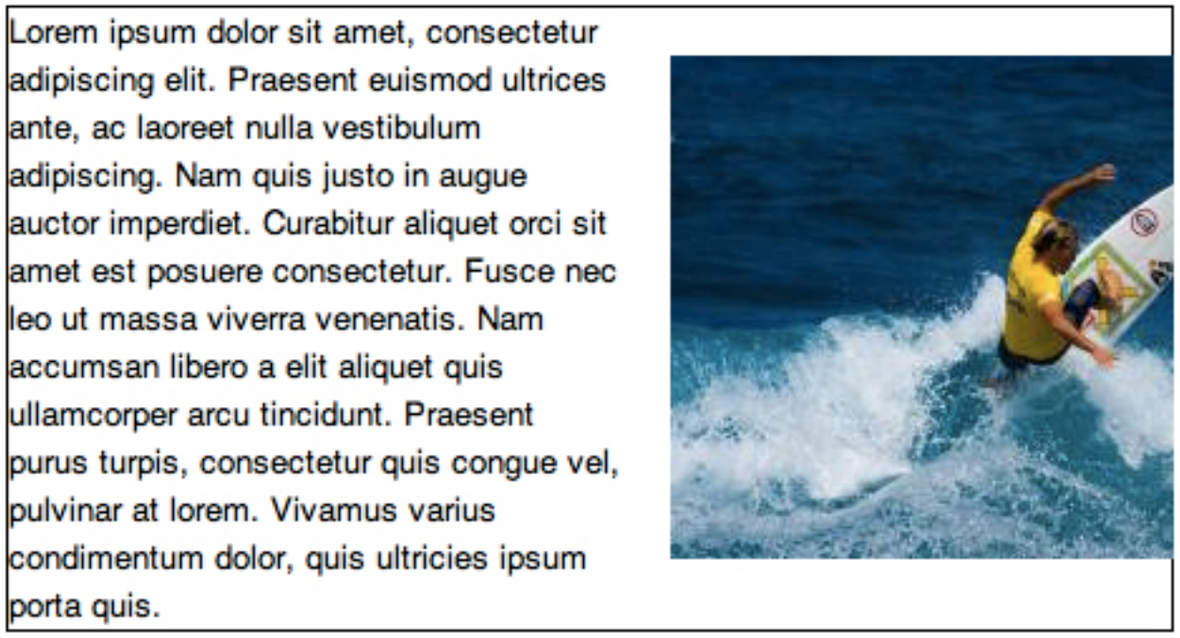
```
img {
  float: right;
  margin: 20px;
}
```



Float: пример

Добавим отступ картинки от текста и границу

```
p {
  margin: 20px;
  border: solid 1px black;
}
```



- Картинка оказалась внутри абзаца
- margin не отработал как ожидалось

Float: пример

Выравниваем абзац слева и указываем ширину блоков

```
img {
  float: right;
  margin: 20px;
}
p {
  float: left;
  width: 220px;
  margin: 20px;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent euismod ultrices ante, ac laoreet nulla vestibulum adipiscing. Nam quis justo in augue auctor imperdiet. Curabitur aliquet orci sit amet est posuere consectetur. Fusce nec leo ut massa viverra venenatis. Nam accumsan libero a elit aliquet quis ullamcorper arcu tincidunt. Praesent purus turpis, consectetur quis congue vel, pulvinar at lorem. Vivamus varius condimentum dolor, quis ultricies ipsum porta quis.

Альтернативный фикс:

```
.clearfix {
  clear: both;
}
```

Float: пример

Основная проблема – блок вырван из потока документа

Поток документа – порядок отображения элементов на странице. По умолчанию порядок отображения HTML-элементов в браузере соответствует порядку описания элементов в исходном коде.

В нормальном потоке документа элементы располагаются последовательно, каждый с новой строки – для блочных элементов, в одной строке – для строчных элементов.

Float: ссылки

MDN: Базовый поток

HTMLbook: вывести элемент из потока

MDN: Introduction to CSS layout

Поток документа. Свойства float и clear

MDN: Блочный контекст форматирования

inline-block – блочно-строчный каркас

- `display: inline-block`
- Для блоков строчных элементов
- Ведет себя как текст
- Почти flex для старых браузеров

inline-block – проблемы с текстом

```
<ul>
  <li>one</li>
  <li>two</li>
  <li>three</li>
  <li>four</li>
  <li>five</li>
</ul>
```

display: inline-block

one two three four five

float: left

one two three four five

display: inline-block, no whitespace in HTML

one two three four five

```
<ul>
  <li>one</li><li>two</li><li>three</li><li>four</li><li>five</li>
</ul>
```

inline-block – как решить проблему

- `font-size: 0` у родителя
- Убрать пробелы в разметке
- ~~Отрицательный margin~~
- ~~Удалить пробелы из JavaScript~~

```
<ul>
  <li>one</li>
  <li>two</li>
  <li>three</li>
  <li>four</li>
  <li>five</li>
</ul>
```

display: inline-block

one two three four five

float: left

one two three four five

display: inline-block, no whitespace in HTML

one two three four five

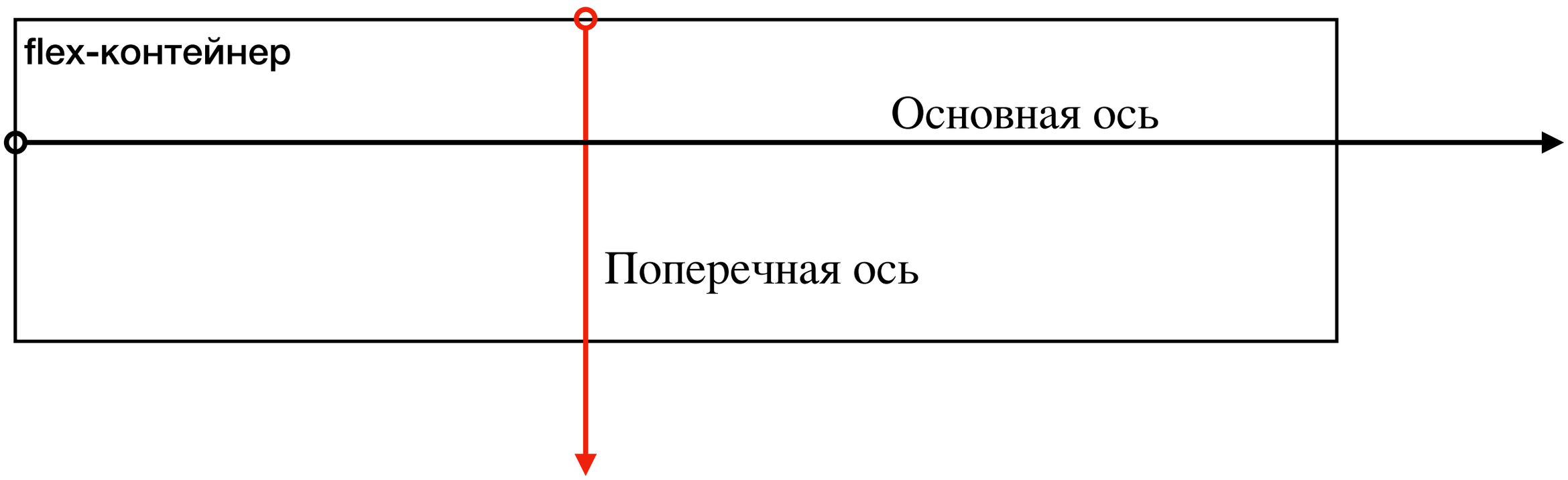
```
<ul>
  <li>one</li><li>two</li><li>three</li><li>four</li><li>five</li>
</ul>
```

Статья про inline-block

Flexbox

`display: flex`

Применяется к: родительскому элементу flex-контейнера



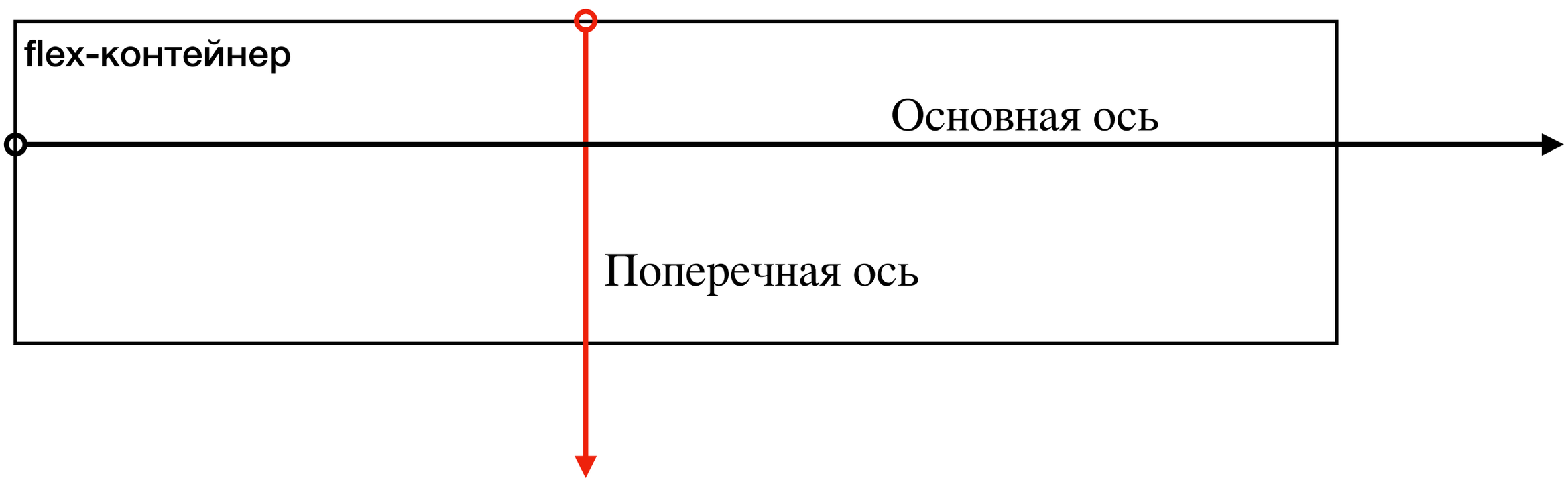
Flexbox

flex-direction: row
row-reverse
column
column-reverse

Устанавливает основную ось, таким образом определяет направление элементов расположенных в контейнере

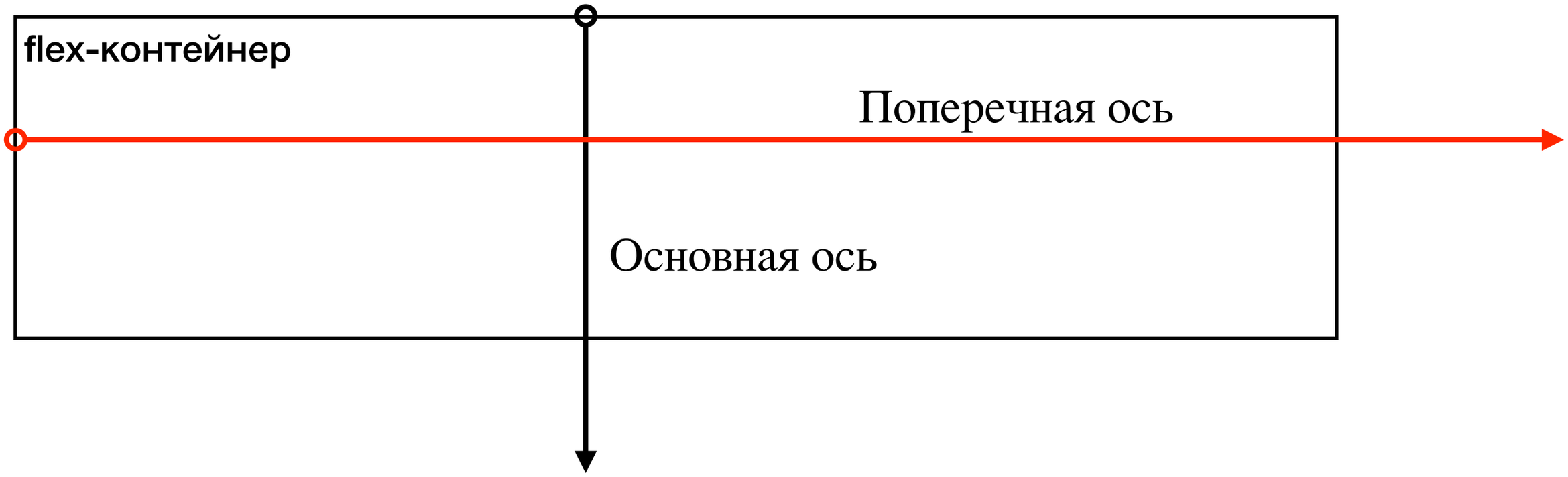
Flexbox

```
flex-direction: row
```



Flexbox

```
flex-direction: column
```

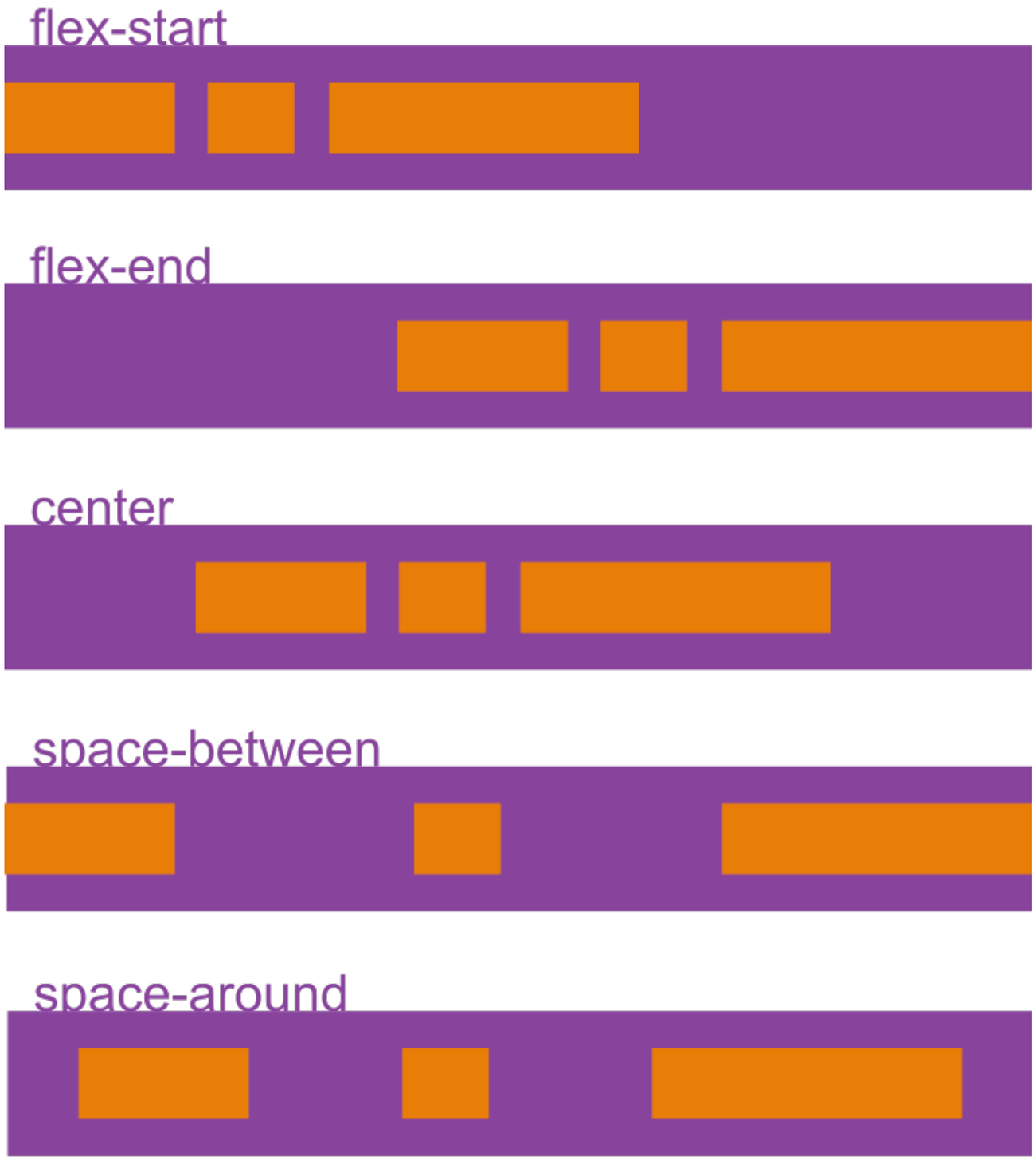


Flexbox

`justify-content:`

`flex-start`
`flex-end`
`center`
`space-between`
`space-around`

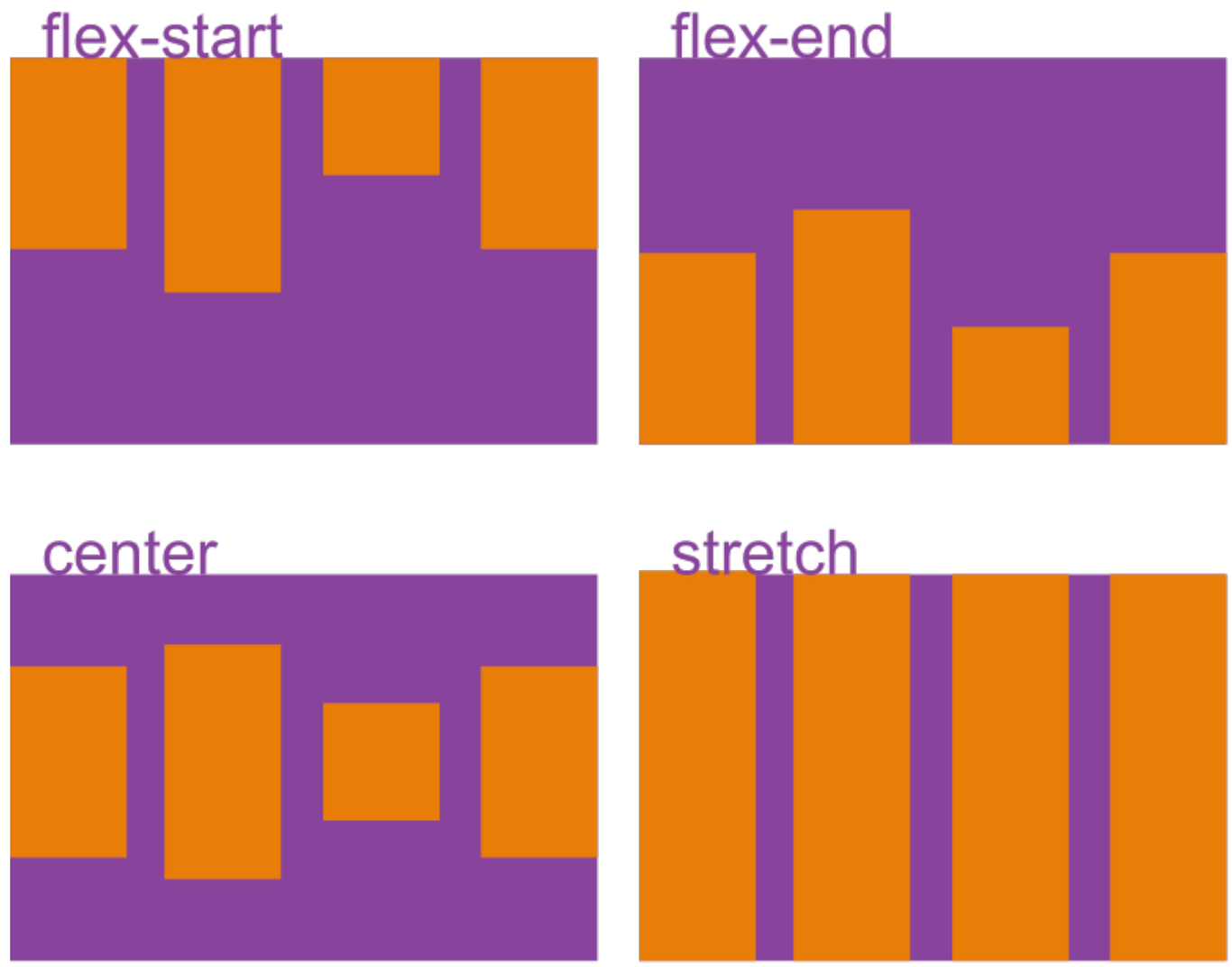
Определяет выравнивание вдоль основной оси



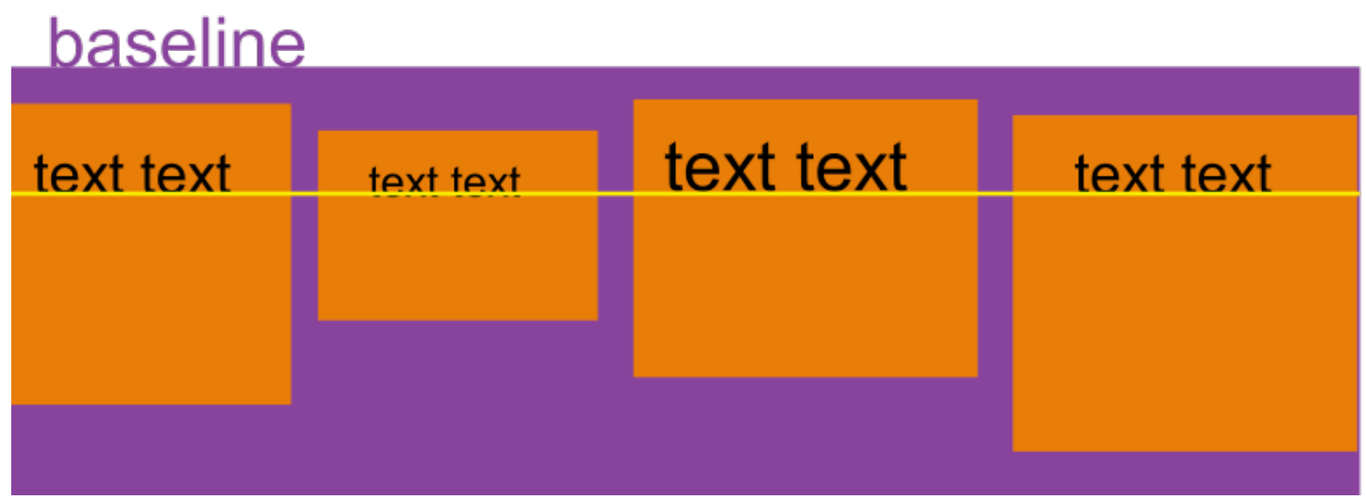
Flexbox

`align-items:`

`flex-start`
`flex-end`
`center`
`stretch`
`baseline`



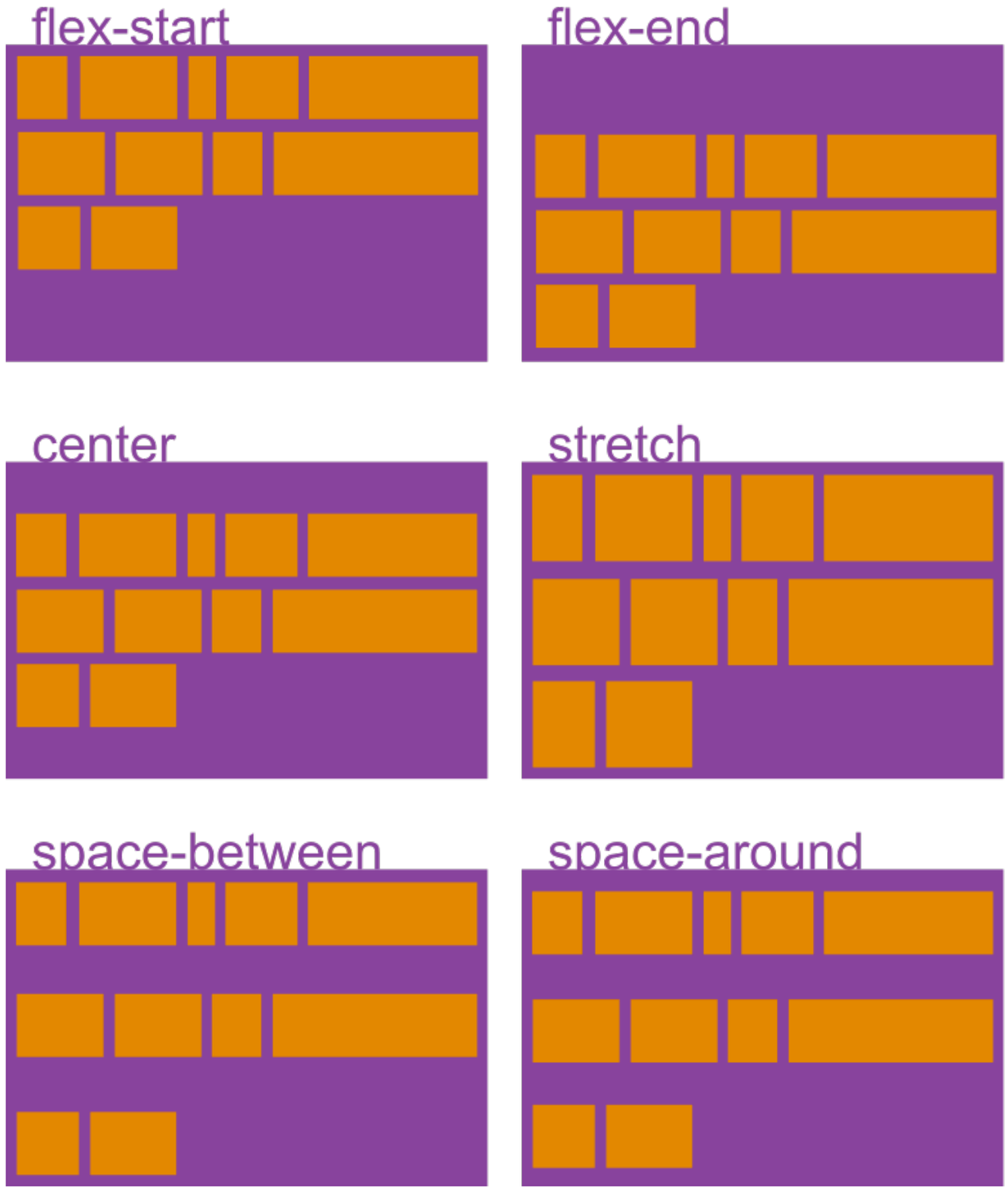
Определяет выравнивание вдоль поперечной оси на текущей строке



Flexbox

`align-content:`

`flex-start`
`flex-end`
`center`
`space-between`
`space-around`



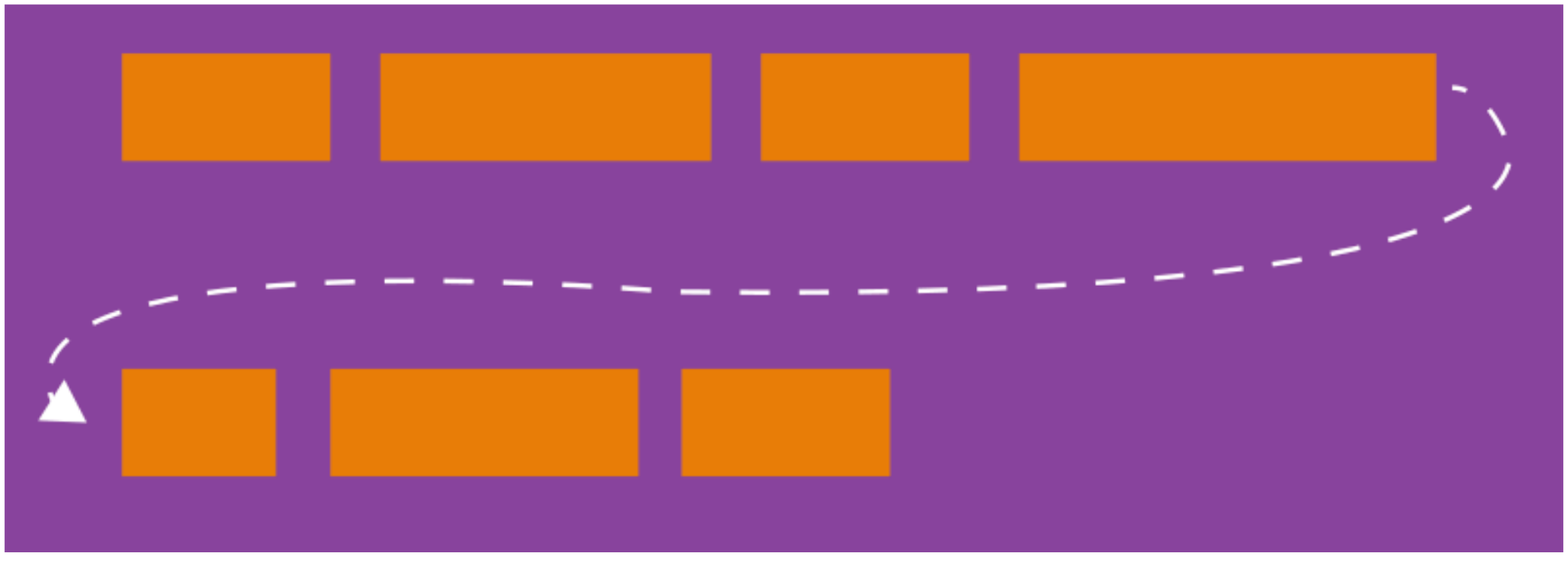
Выравнивает строки flex-контейнера при наличии свободного места на поперечной оси аналогично тому, как это делает `justify-content` на главной оси

Применяется к родительскому элементу flex-контейнера

Flexbox

`flex-wrap:` `nowrap`
`wrap`
`wrap-reverse`

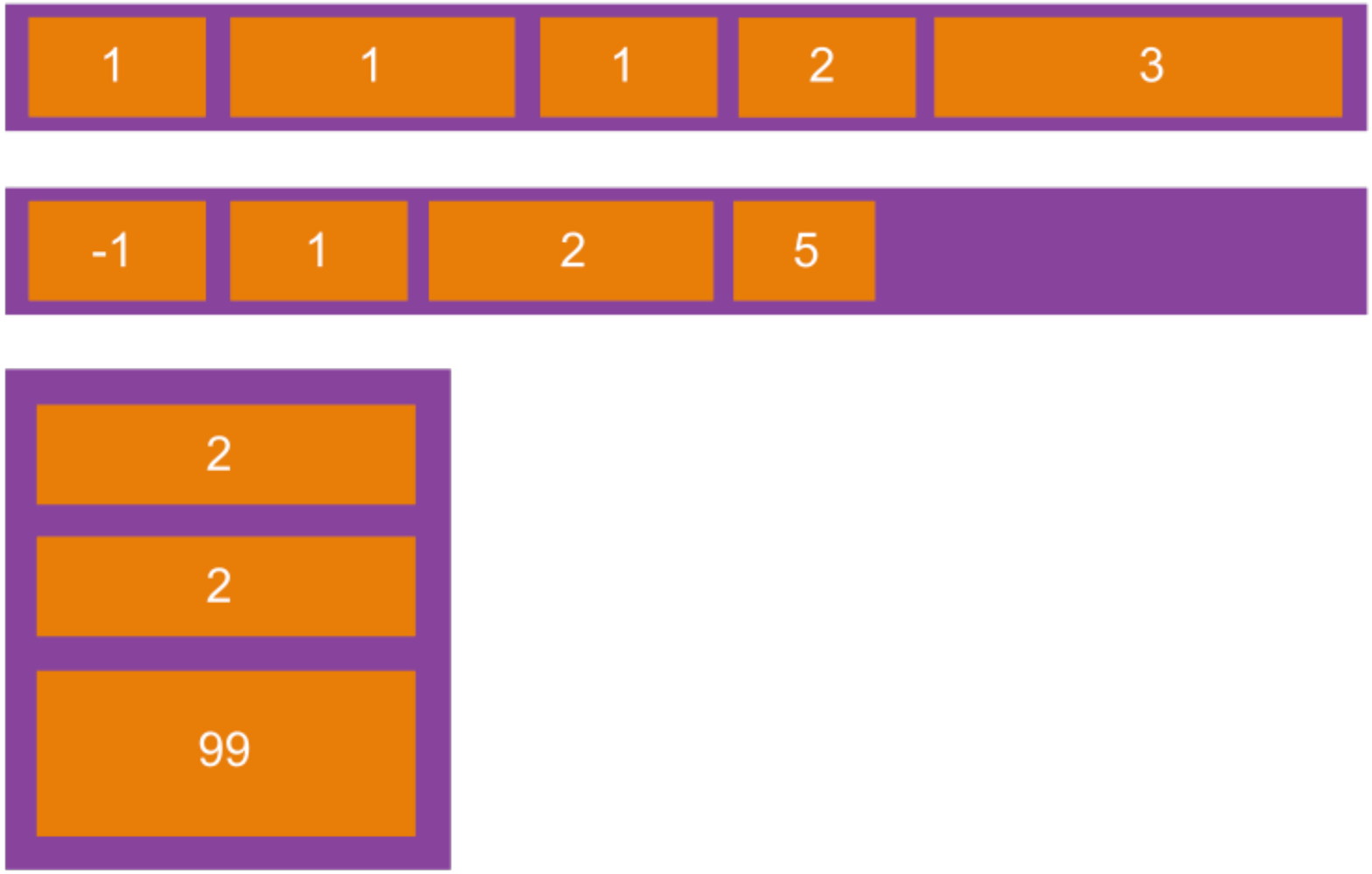
По умолчанию, элементы будут пытаться заполнить только одну строку.
Свойство позволяет элементам переноситься на следующую строку, при необходимости.



Flexbox

`order:` `число`

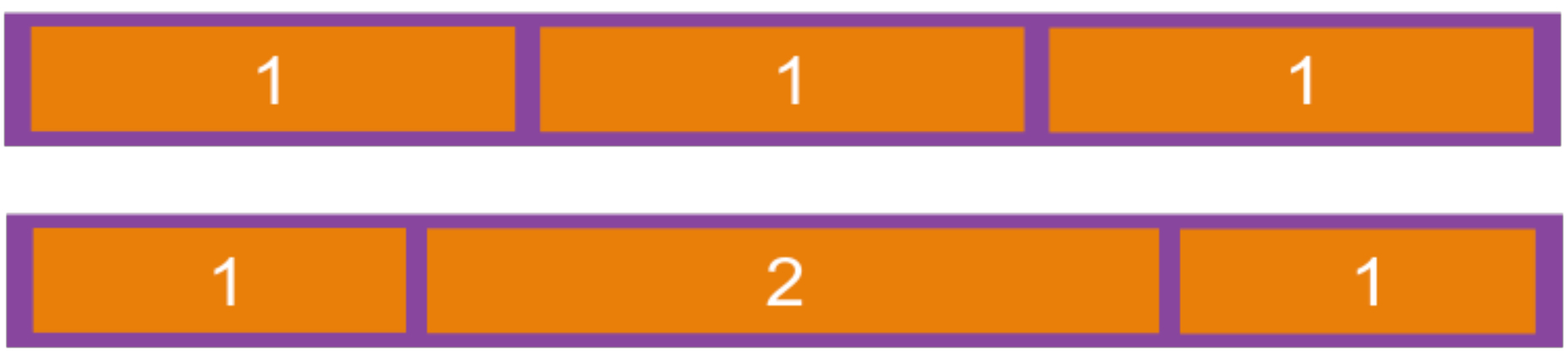
По умолчанию, все элементы располагаются в исходном для них порядке.
Данное свойство управляет порядком, в котором располагаются элементы внутри контейнера



Flexbox

`flex-grow:` `число`

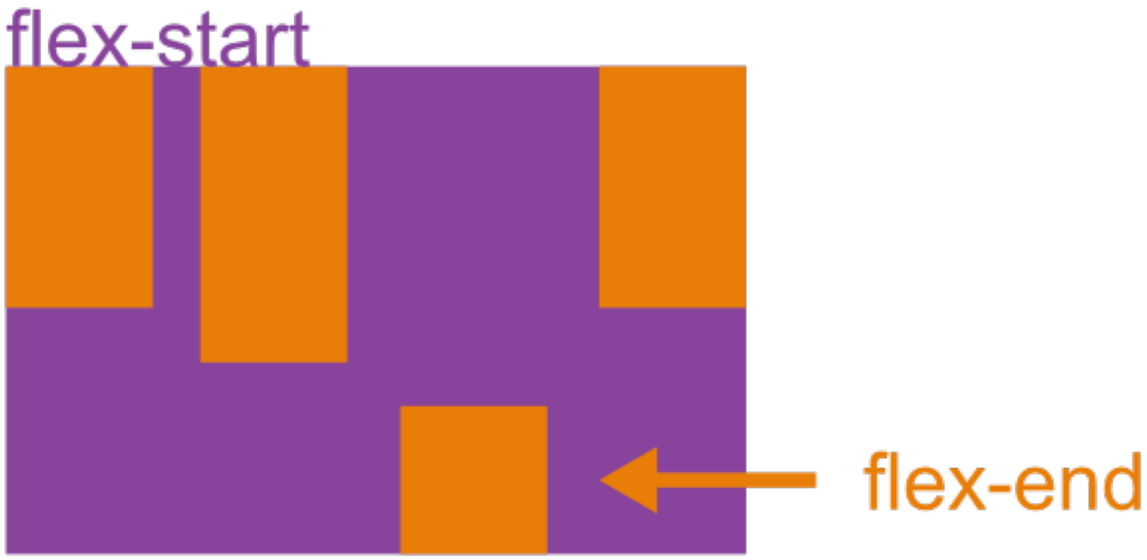
Определяет возможность элемента увеличиваться в размере, при необходимости
Определяет какое количество свободного пространства (в пропорции) внутри контейнера должен занимать элемент



Flexbox

`align-self:`

`auto`
`flex-start`
`flex-end`
`center`
`stretch`
`baseline`



Позволяет переопределить выравнивание по умолчанию (или заданное с помощью свойства align-items) для отдельных flex-элементов

Flexbox: ссылки

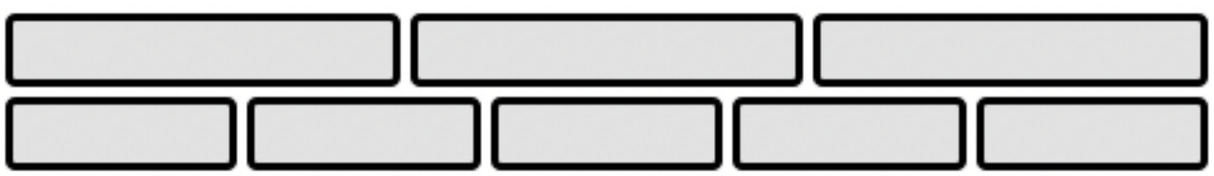
Frontender Magazine: Полное руководство по Flexbox

A Complete Guide to Flexbox

MDN: Flexbox

Flexbox Froggy 🎮 🐸

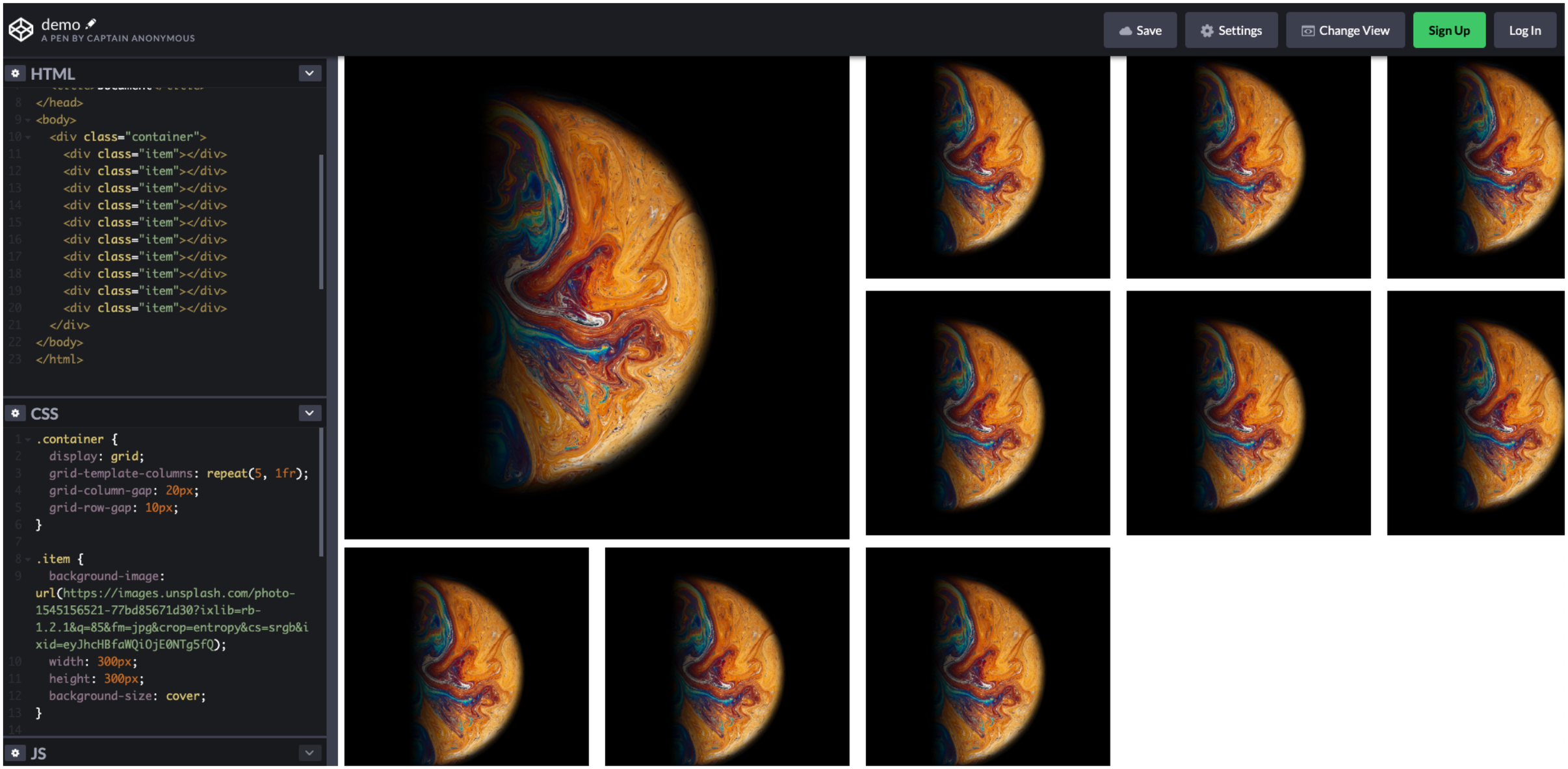
Flex vs Grid Layout



Flex

Grid Layout

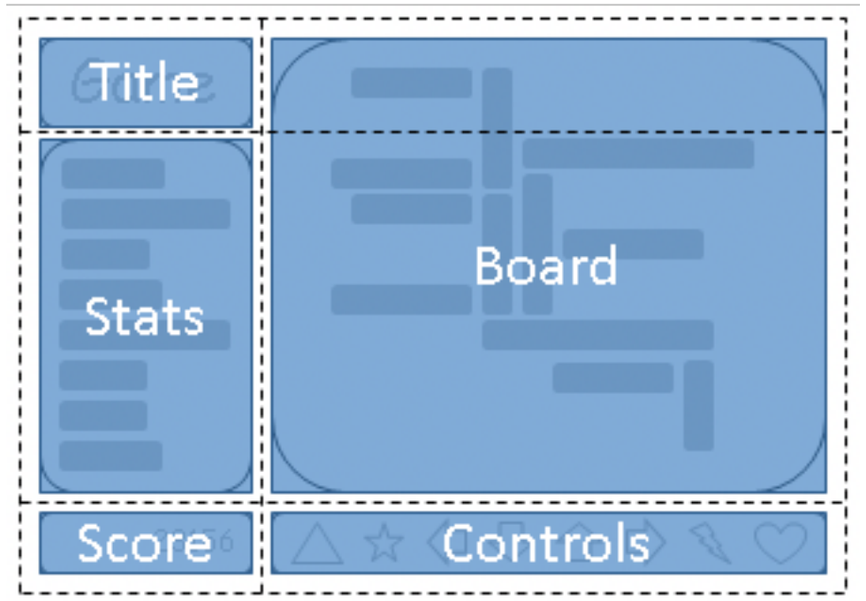
Grid Layout


demo
A PEN BY CAPTAIN ANONYMOUS
Save
Settings
Change View
Sign Up
Log In
HTML
</head>
<body>

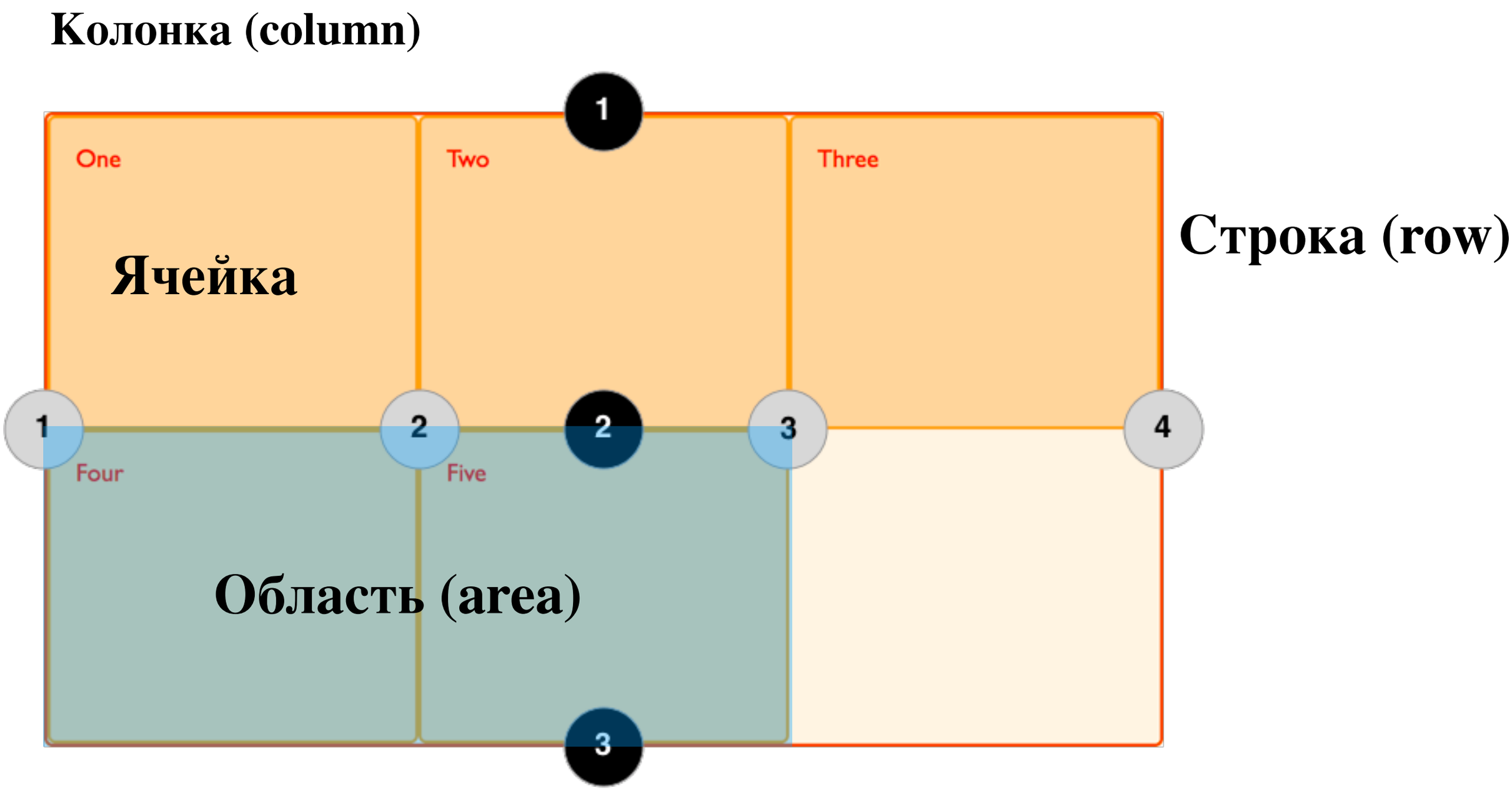
</body>
</html>
CSS
.container {
display: grid;
grid-template-columns: repeat(5, 1fr);
grid-column-gap: 20px;
grid-row-gap: 10px;
}
.item {
background-image:
url(https://images.unsplash.com/photo-1545156521-77bd85671d30?ixlib=rb-1.2.1&q=85&fm=jpg&crop=entropy&cs=srgb&ixid=eyJhcHBfaWQiOjE0NTg5fQ);
width: 300px;
height: 300px;
background-size: cover;
}
JS

Grid Layout

- `display: grid;`
- Как таблицы, но лучше
- Для каркаса страницы



Grid Layout


Колонка (column)
1
One
Two
Three
Ячейка
Строка (row)
1
2
2
3
4
Four
Five
Область (area)
3

Grid Layout

http://japanese-grid.webflow.io/

Grid Layout: пример

```
<div class="container">
  <div class="logo">Logo</div>
  <nav class="menu-horizontal">Menu_1</nav>
  <nav class="menu-vertical">Menu_2</nav>
  <div class="main-text">Text Content</div>
  <div class="banner-zone">Banners</div>
  <div class="copyright">Copyright</div>
  <div class="social">Social Links</div>
</div>
```

```
div, nav {
  border: 1px dotted grey;
}
.container {
  display: grid;
  height: 100vh;
  grid-template-columns: repeat(3, 1fr);
  grid-template-rows: 50px 1fr 50px;
  grid-template-areas:
    "logo logo menu-horizontal"
    "menu-vertical main-text banner-zone"
    "copyright copyright social";
}
.logo {
  grid-area: logo;
}
.menu-horizontal {
  grid-area: menu-horizontal;
}
.menu-vertical {
  grid-area: menu-vertical;
}
.main-text {
  grid-area: main-text;
}
.banner-zone {
  grid-area: banner-zone;
}
.copyright {
  grid-area: copyright;
}
.social {
  grid-area: social;
}
```

https://jsfiddle.net/3e9jhg0y/

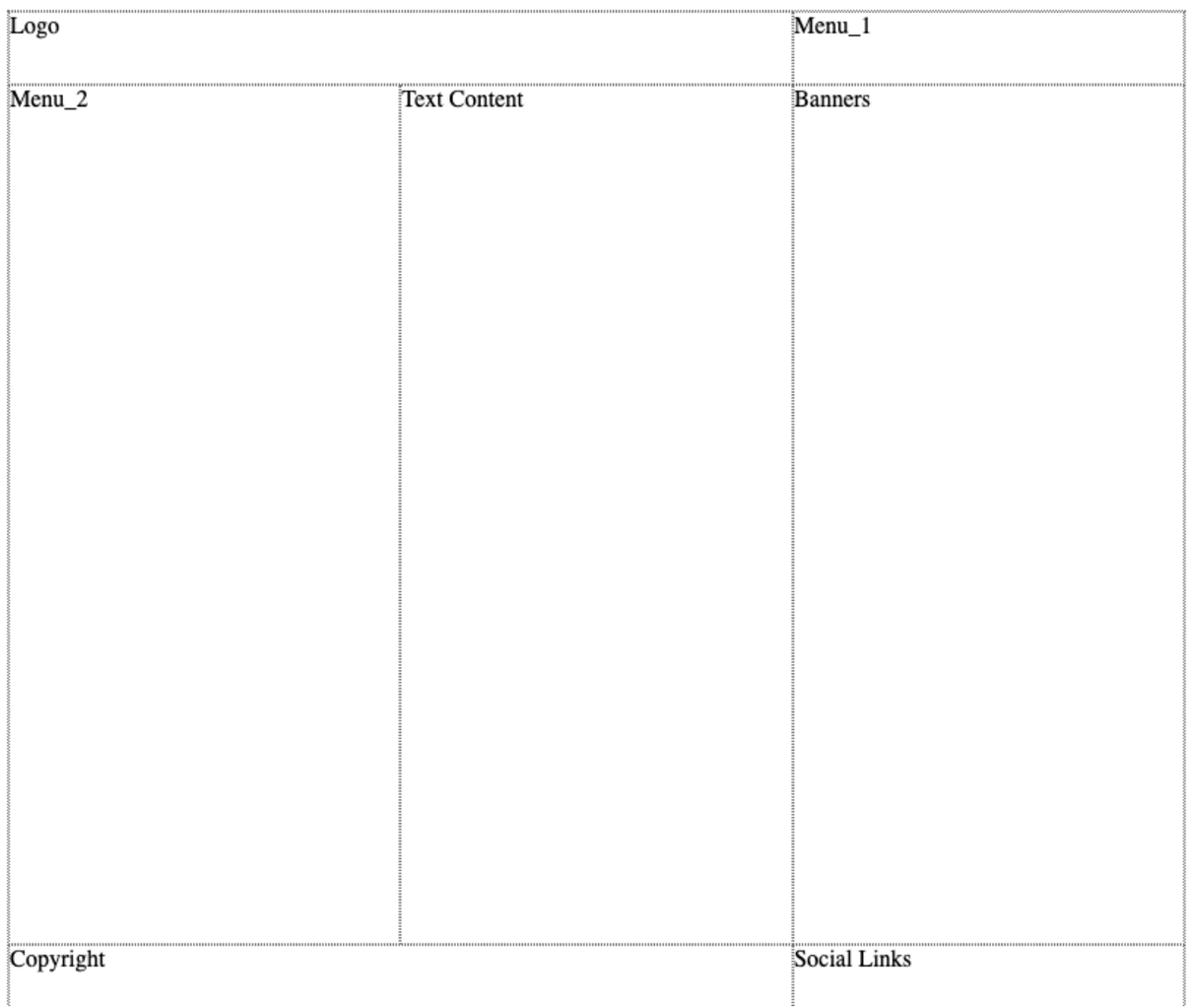
Grid Layout: пример

Logo		Menu_1
Menu_2	Text Content	Banners
Copyright		Social Links

Grid Layout: пример

https://jsfiddle.net/3e9jhg0y/

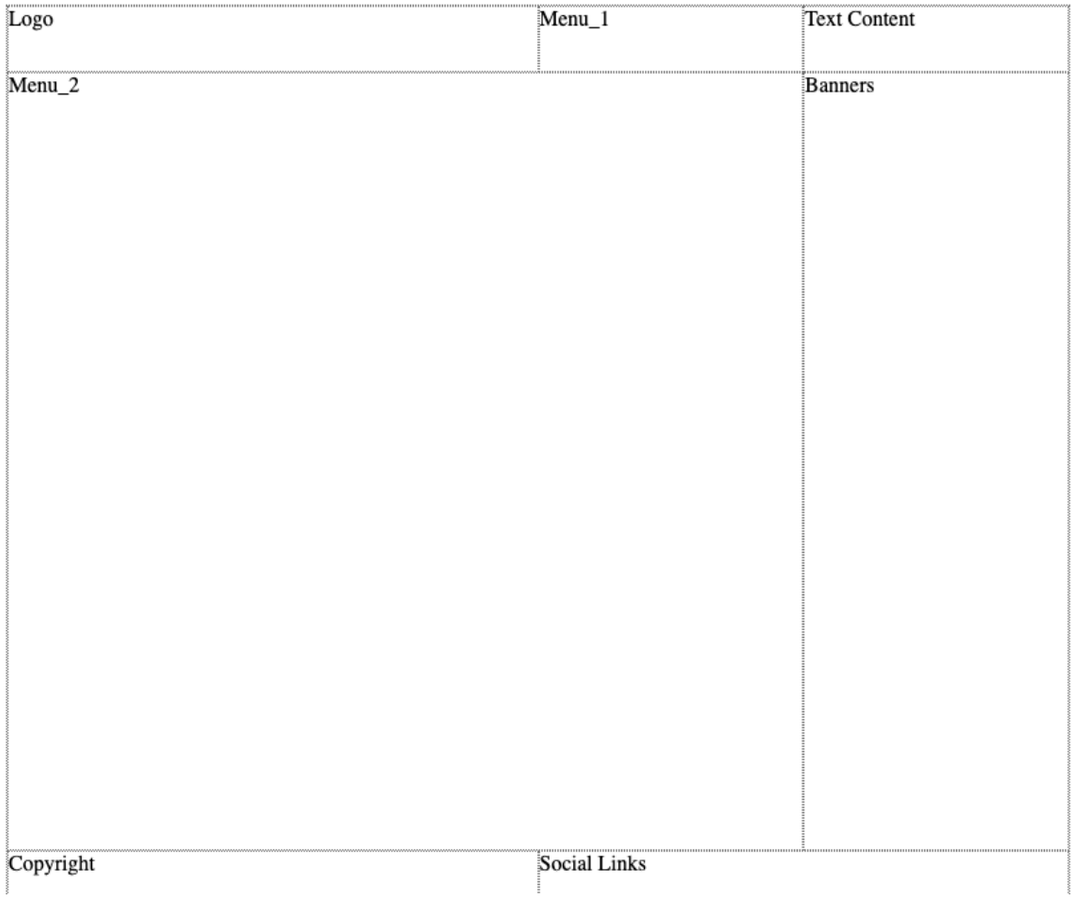
```
.container {
  display: grid;
  height: 100vh;
  grid-template-columns: repeat(3, 1fr);
  grid-template-rows: 50px 1fr 50px;
  grid-template-areas:
    "logo logo menu-horizontal"
    "menu-vertical main-text banner-zone"
    "copyright copyright social";
}
```



Grid Layout: пример

https://jsfiddle.net/g5zajt3m/2/

```
.container {
  display: grid;
  height: 100vh;
  grid-template-columns: repeat(4, 1fr);
  grid-template-rows: 50px 1fr 50px;
  grid-template-areas:
    "logo logo menu-horizontal main-text"
    "menu-vertical menu-vertical menu-vertical banner-zone"
    "copyright copyright social social";
}
```



Grid Layout: пример

https://jsfiddle.net/xstvcw9p/2/

```
.container {
  display: grid;
  height: 100vh;
  grid-template-columns: repeat(4, 1fr);
  grid-template-rows: 50px 1fr 50px;
  grid-template-areas:
    "logo logo menu-horizontal main-text"
    "menu-vertical menu-vertical menu-vertical
banner-zone"
    "copyright copyright social social";
}

@media screen and (max-width: 350px) {
  .container {
    grid-template-columns: 1fr;
    grid-template-rows: auto;
    grid-template-areas:
      "banner-zone"
      "main-text"
      "menu-horizontal"
      "logo"
      "social"
      "menu-vertical"
      "copyright";
  }
}
```

Grid Layout: ссылки

[A Complete Guide to Grid Layout]

[MDN: Grid Layout]

[CSS Grid Garden] 🎮 🥕

Что в итоге?

- table только для таблиц
- float для обтекания текстом
- inline-block для блочности строчных элементов
- flex для блоков в строку/столбец
- grid для построения сетки сайта

Не забываем о поддержке

CSS Grid Layout (level 1) - CR

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Usage	% of all users	
Global	91.61% + 1.96% =	93.57%
unprefixed:	91.61%	

Current aligned | Usage relative | Date relative | Apply filters | Show all | ?

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Blackberry Browser	Opera Mobile *	Chrome for Android	Firefox for Android	IE Mobile	UC Browser for Android	Samsı Interr
		2-39	4-28												
		3 40-51	1 29-56		10-27										
6-9	2 12-15	4 52-53	4 57	3.1-10	1 28-43	3.2-10.2									4-5
2 10	16-17	54-68	58-76	10.1-12	44-60	10.3-12.1		2.1-4.4.4	7	12-12.1			2 10		6.2-
2 11	18	69	77	12.1	62	12.3	all	76	10	46	76	68	2 11	12.12	9.2
	76	70-71	78-80	13-TP		13									

Can I use?

Спасибо!