

Яндекс

CSS

Занятие 1

Павел Пушкарь, разработчик интерфейсов

HTML + CSS

- HTML + CSS = Веб-страница
- **HTML** отвечает за структуру и содержание
- **CSS** отвечает за внешний вид



HTML

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>Яндекс.Ясли</title>
</head>
<body>
  <!--Привет, Мир!-->
  <h1>Hello, World!</h1>
</body>
</html>
```

Что такое CSS?

- **Cascading Style Sheets** - каскадные таблицы стилей
- CSS - язык, который используется для оформления внешнего вида HTML-документов

```
<head>
  <meta charset="UTF-8">
  <title>Яндекс.Ясли</title>
  <style>
    h1 {
      color: red;
    }
  </style>
</head>
```

CSS-правило

Стили страницы формируются с помощью списка CSS-правил

```
селектор {  
  свойство: значение;  
  свойство: значение;  
  свойство: значение;  
  ...  
}
```

```
.b-page__body {  
 margin: ▶ 0;  
 color: ■ #000;  
 background: ▶ □ #fff;  
 height: 100%;  
 padding: ▶ 0;  
 font-family: Arial, Helvetica, sans-serif;  
 font-size: 13px;  
 line-height: 1.154;  
}
```

Селекторы

```
<body>  
  <h1>Hello, World!</h1>  
  <p>Параграф 1</p>  
  <p>Параграф 2</p>  
  <p>Параграф 3</p>  
</body>
```

Hello, World!

Параграф 1

Параграф 2

Параграф 3

- Селекторы определяют, к какому элементу применить CSS-правило.

Виды селекторов - различные варианты обращения к элементу



Селектор тега

```
<body>  
  <h1>Hello, World!</h1>  
  <p>Параграф 1</p>  
  <p>Параграф 2</p>  
  <p>Параграф 3</p>  
</body>
```

Параграф 1

Параграф 2

Параграф 3

```
p {  
  font-style: italic;  
}
```

Селектор класса

```
<body>  
  <h1>Hello, World!</h1>  
  <p class="red">Параграф 1</p>  
  <p class="green">Параграф 2</p>  
  <p class="blue">Параграф 3</p>  
</body>
```

```
.red {  
  color: red;  
}  
.green {  
  color: green;  
}  
.blue {  
  color: blue;  
}
```

Параграф 1

Параграф 2

Параграф 3

Контекстные(вложенные) селекторы

```
<ul class='web'>
  <li>HTML</li>
  <li>CSS</li>
  <li>JS</li>
</ul>
```

- HTML
- CSS
- JS

```
<ul class='server'>
  <li>PHP</li>
  <li>NODE</li>
</ul>
```

- PHP
- NODE

```
.web li {
  text-decoration: underline;
}
```

Соседние селекторы

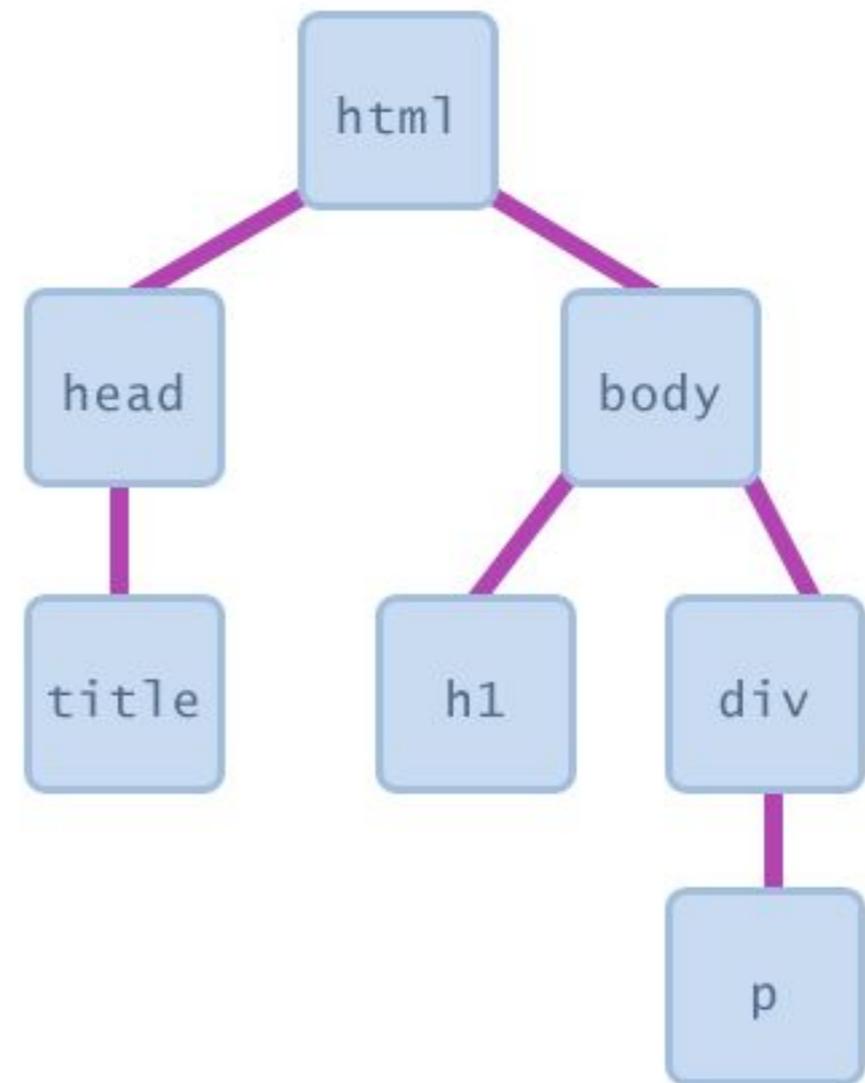
```
<ul class="parent">  
  <li class="item first">1 этаж</li>  
  <li class="item">2 этаж</li>  
  <li class="item">3 этаж</li>  
  <li class="item">4 этаж</li>  
</ul>
```

.first + .item {...} - расположены рядом

.parent .first + .item {...} - вложенный и соседний

Селекторы по родственным СВЯЗЯМ

- **Потомки:**
html p {...},
body p
- **Дочерние** (ближайшие потомки):
div > p, body > div > p
(удобно использовать с многоуровневыми списками)



Псевдоклассы

```
<ul class='web'>
  <li>HTML</li>
  <li>CSS</li>
  <li>JS</li>
  <li>PHP</li>
  <li>NODE</li>
</ul>
```

HTML

CSS

JS

PHP

NODE

```
li:nth-child(2n) {
  background-color: #4eb543;
}
li:nth-child(2n+1) {
  background-color: #f2f0fd;
}
```

Псевдоклассы для ссылок

- **a:link { ... }** - не посещенные ссылки

- **a:visited { ... }** - посещенные ссылки

 **Разработка интерфейсов** — Обучение в Яндексе

 **Разработка интерфейсов** — Обучение в Яндексе

 **Разработка интерфейсов** — Обучение в Яндексе

target="_blank">

class="article-item" >селекторы

По классу: .article-link {...}

По имени тега: a {...}

По идентификатору: #main-link {...}

По любому атрибуту: [target="_blank"] {...} или
[class="article-link"] {...}

(чаще всего используются в формах)

Селекторы можно комбинировать, уточняя выбор

a.article-link {...}

.article-link .article-item {...} -

контекстные/вложенные

Итого

- HTML+CSS = web-страница
- HTML = структуру и содержание
- CSS = внешний вид
- CSS формируется списком CSS-правил
- Селекторы - используем для поиска элементов и применения к ним стилей

Загадка?

```
<p class="hot cold">  
  Синий или красный?  
</p>
```

```
.cold {  
  color: blue;  
}  
  
.hot {  
  color: red;  
}
```



Каскадность

Каскадность

- **Каскадность** - к одному элементу могут применяться несколько CSS-правил (свойств)
- **Конфликты свойств**
- **Правила для разрешения конфликтов:**
 - Специфичность
 - Наследование

```
<p class="hot cold">  
  Синий или красный?  
</p>  
  
.cold {  
  color: blue;  
}  
  
.hot {  
  color: red;  
}
```

Специфичность

Чем **меньше** количество элементов попадает под данный селектор, тем он **специфичней**

Расчет специфичности:

- #id —> 100
- .class —> 10
- tag —> 1

Селектор
<i>div</i>
<i>div .cold</i>
<i>#super .cold</i>
<i>div span</i>
<i>.cold</i>
<i>.cold .blue</i>
<i>#super span</i>

#super .cold VS #super span

Наследование

- **Наследование** - механизм, с помощью которого значения свойств элемента-родителя передаются его элементам-потомкам

```
<p class="text">  
  Строка с выделенным <span>словом</span>  
</p>
```

```
<style>  
.text{  
  color: red;  
}  
</style>
```

Если бы не было наследования:

Строка с выделенным словом

Строка с выделенным словом

Наследуемые СТИЛИ

- **Наследуются** - параметры отображения текста (размер шрифта, тип шрифта, цвет и т.д.)
font-size, font-family, font-style, font-weight, color...

Например: стили шрифта для всего документа задаем в body {...}

<https://www.w3.org/TR/CSS21/propidx.html>

Не все стили наследуются

```
.text-border{  
  width: 260px;  
  border: 1px solid red;  
}
```

Строка с выделенным словом

```
<p class="text-border">  
  Строка с выделенным <span>словом</span>  
</p>
```

Если бы свойство наследовалось:

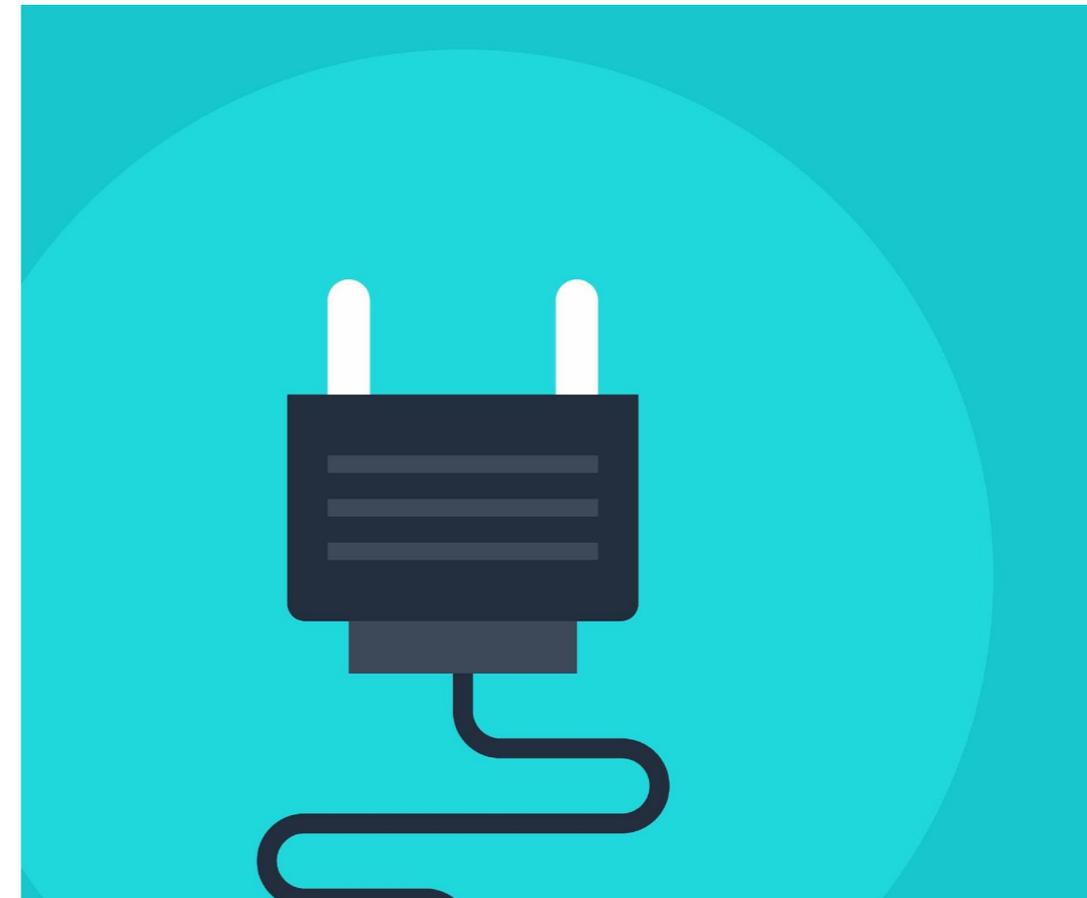
Строка с выделенным **словом**

Не наследуются

- **Не наследуются** - размеры, отступы, фоны, рамки -
background, border, padding, margin, width, height...

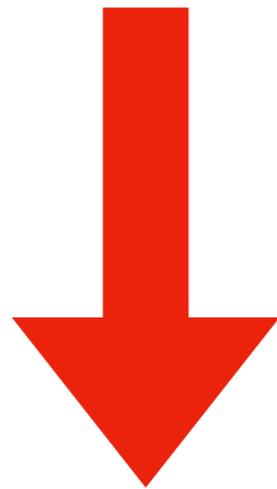
Способы подключения CSS

- `<style>...</style>`
- `<link rel="stylesheet" href="style.css"/>`
- `<tag style="color:red;">`
....
`</tag>`



Что в итоге?

- Каскадность, наследование, специфичность
- Способы подключения CSS



Итоговый вид элемента

```
<p class="hot cold">
  Синий или красный?
</p>

.cold {
  color: blue;
}

.hot {
  color: red;
}
```

Сейчас в СМИ

Актуальные новости и события дня

[USD MOEX](#) курс MOEX на 01/0365,94+0,03
+0,03

[EUR MOEX](#) курс MOEX на 01/0374,88-0,12
-0,12

[Нефть](#) цена на 01/0365,01-2,05%
-2,05%



Кэшбэк 5% баллами при оплате Я.Кардом

Узнайте подробности 0+

[Видео](#)[Картинки](#)[Новости](#)[Карты](#)[Маркет](#)[Переводчик](#)[Музыка](#)[Эфир](#)[ещё](#)

Найти

Найдётся всё. Например, мультфильмы



ВЖУХ



Симферополь

Перейти на Яндекс.Украина Настройка

Сейчас в СМИ в Республике Крым 2 марта, суббота 17 50

- Опубликованы кадры встречи российского фрегата с американским эсминцем
- Генштаб России объяснил суть американского «Троянского коня»
- Stern: западные эксперты недооценили ракету «Циркон»
- Появилось видео запуска корабля Илона Маска Dragon 2 к МКС
- Александр Емельяненко признался в нетрезвом вождении

USD MOEX 65,94 +0,03 EUR MOEX 74,88 -0,12 НЕФТЬ 65,01 -2,05% ...



Кэшбэк 5% баллами при оплате Я.Кардом
Узнайте подробности 0+

Яндекс

Найти

Найдётся всё. Мультфильмы



Погода

+1°
Вечером -1,
ночью -5

Карта Симферополя

Пробки Такси Расписания

Сейчас в эфире

- Золотая лихорадка МУЗ-ТВ
- Футбол как он есть Футбол
- The World Sailing Show. 7-й... Ocean TV

Посещаемое

Маркет — телевизоры

Карты — адреса блошиных рынков

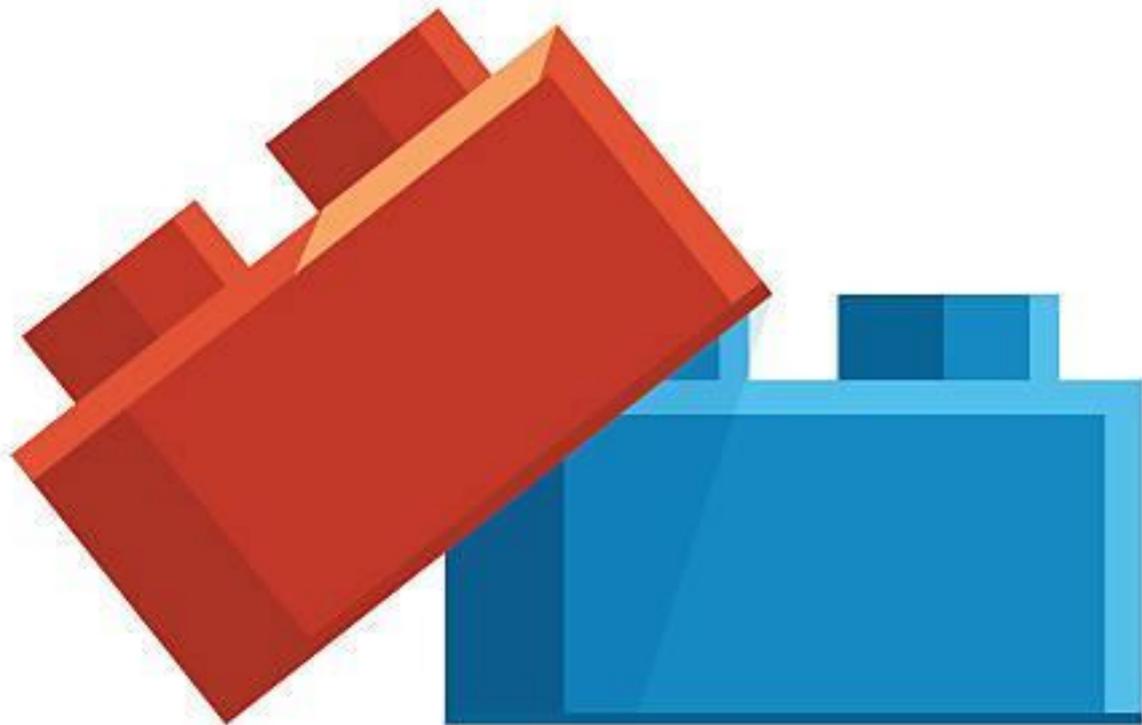
Телепрограмма

17:35 Дело № 306 Культура

17:40 Репортаж Россия 24

17:45 Подарок одинокой женщине

Блочная модель документа



Блочные и строчные

BLOCK:



INLINE:



- HTML страница состоит из вложенных друг в друга прямоугольников
- Блочные элементы
- Строчные элементы

Блочные элементы

- Всегда имеют форму прямоугольника
- По умолчанию тянется на всю ширину родителя
- Высота зависит от содержимого
- Блоку можно задавать размеры и отступы
- До и после блочного элемента есть перенос строки
- К блочным элементам относятся такие теги как: `<div>`, `<p>`, `<h1>`, `<h2>`, ``

Размеры блока

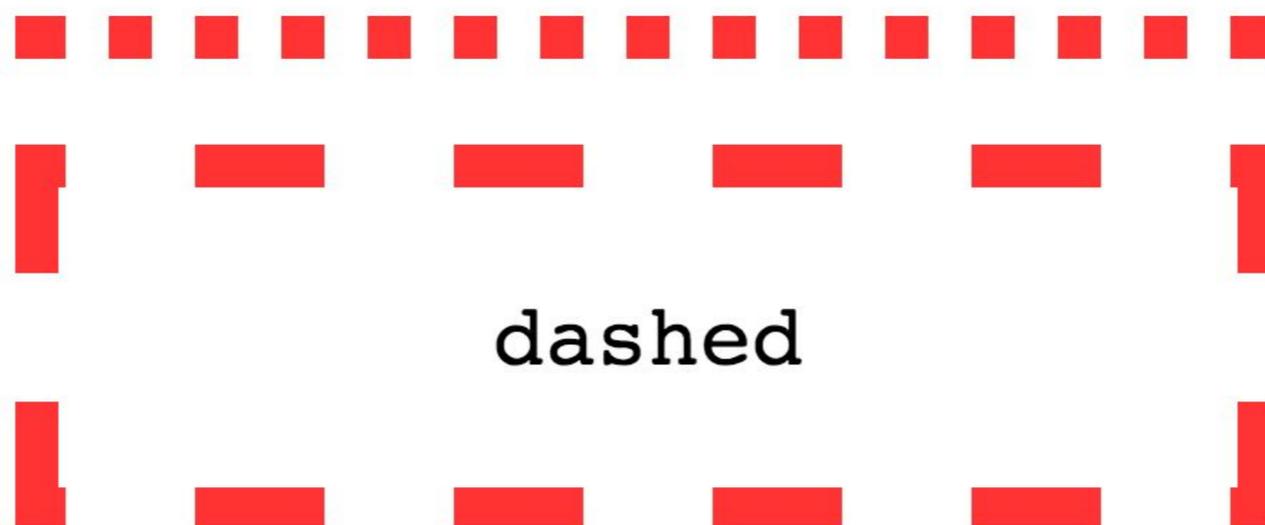
- **width: 50%;** - ширина *содержимого* блока
- **height: 200px;** - высота *содержимого* блока

DANGER! Выпадение!

visible
ЭтотТекстВылезаетСправаЭтотТекстВылезаетСправа
Этот текст вылезает снизу
Этот текст вылезает снизу
~~Этот текст вылезает снизу~~
Этот текст вылезает снизу



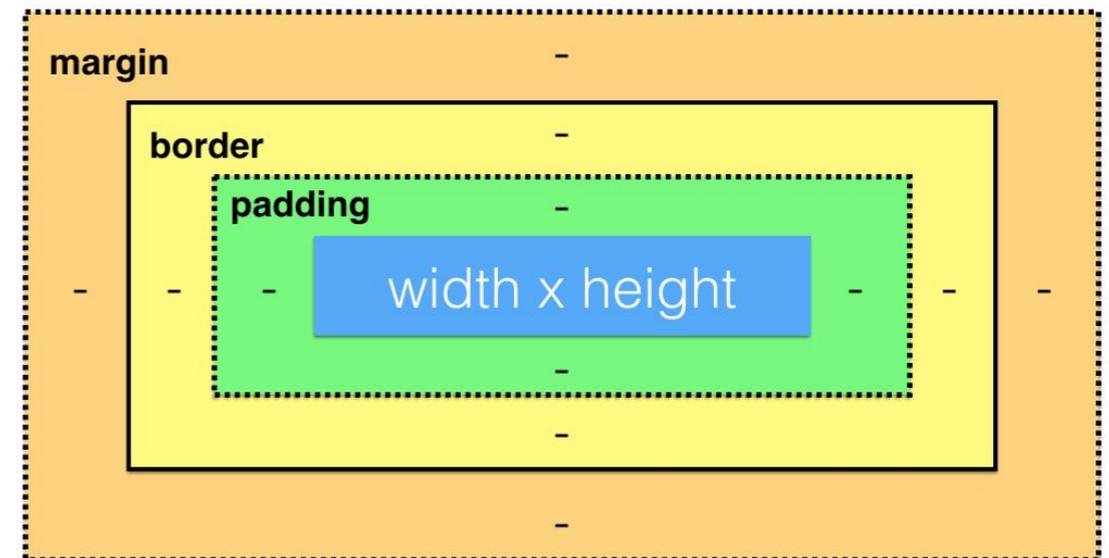
border: 2px solid red;



padding: 10px;

- Внутренний отступ от границы блока до содержимого

```
padding-left: 10px;  
padding-right: 20px;  
padding-top: 30px;  
padding-bottom: 40px;
```



```
padding: 10px; /* одинаково со всех сторон */
```

```
padding: 10px 20px; /* сверху+снизу, справа+слева */
```

```
padding: 10px 20px 30px; /* сверху, справа+слева, снизу */
```

```
padding: 10px 20px 30px 40px; /* сверху, справа, снизу, слева */
```

margin: 10px;

- Внешние отступы между границами блока и его родителем и соседям

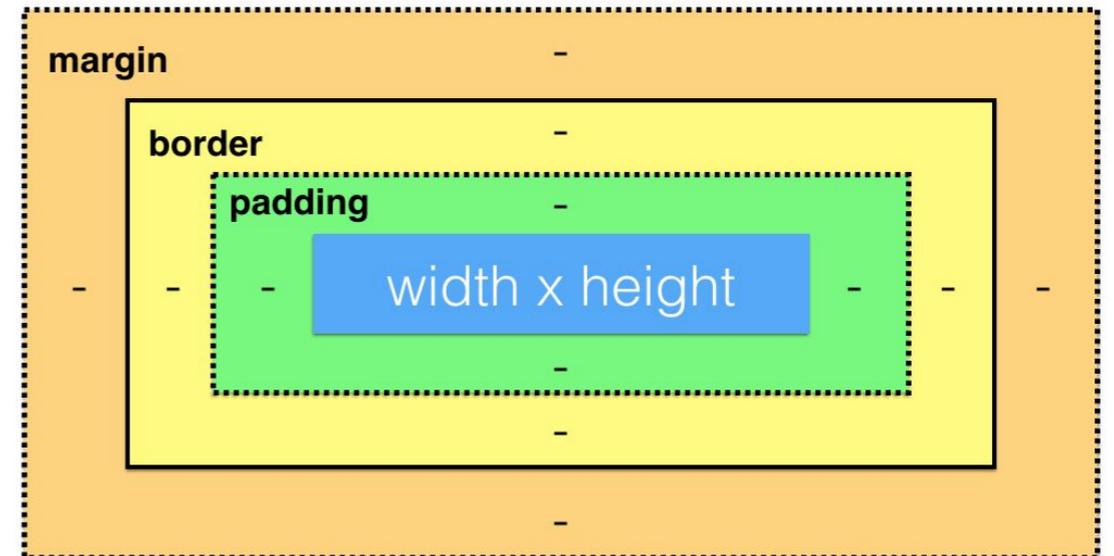
```
margin-left: 10px;  
margin-right: 20px;  
margin-top: 30px;  
margin-bottom: 40px;
```

```
margin: 10px; /* одинаково со всех сторон */
```

```
margin: 10px 20px; /* сверху+снизу, справа+слева */
```

```
margin: 10px 20px 30px; /* сверху, справа+слева, снизу */
```

```
margin: 10px 20px 30px 40px; /* сверху, справа, снизу, слева */
```



Загадка

Схлопывание внешних отступов

Горизонтальный 20px ← 40px → 20px Отступ

Вертикальный 20px

Блочные элементы

К свойствам блочной модели относятся те, которые влияют на размер элемента.

- **width** - ширина блока
- **height** - высота блока
- **border** - рамка
- **padding** - отступ от границы блока до содержимого
- **margin** - отступ между границами блока и его родителем и соседям



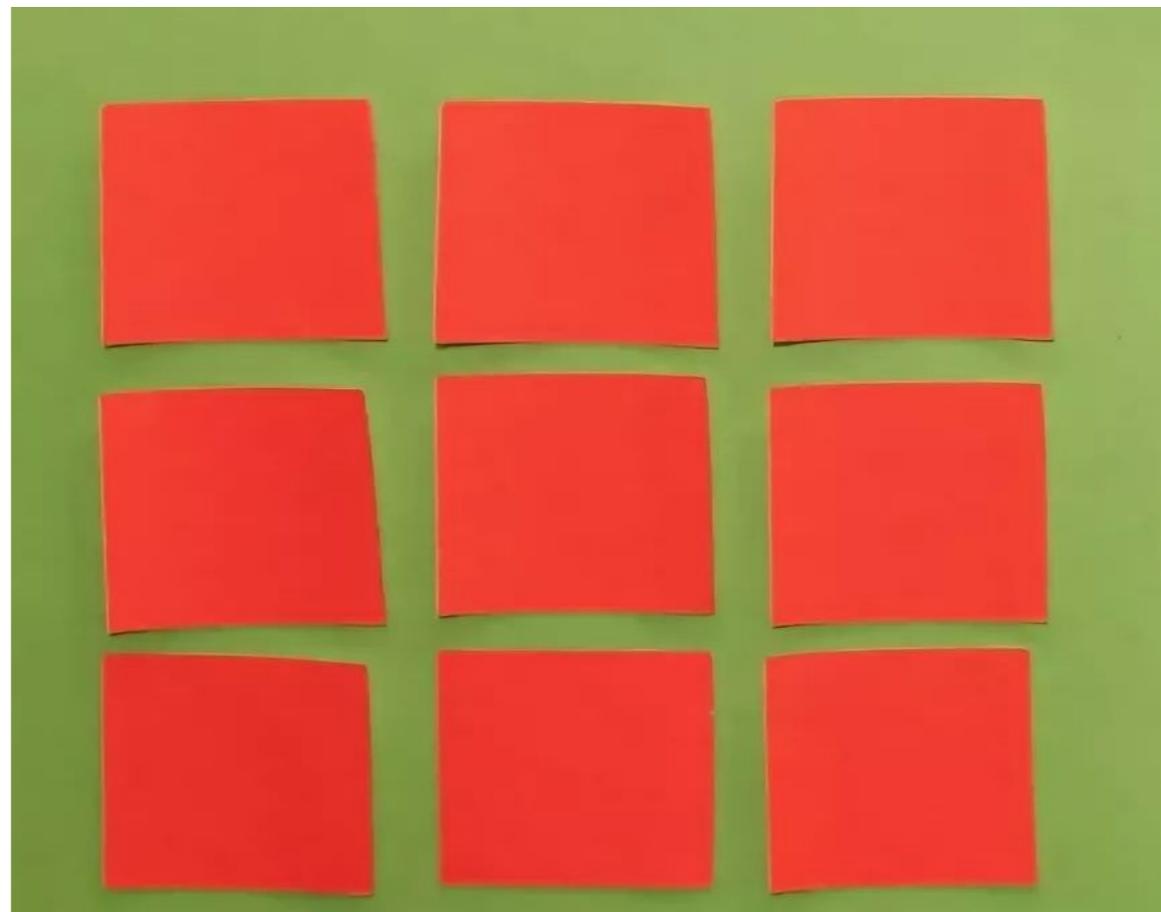


Инлайновые элементы

Инлайновые элементы

- Инлайновые = строчные
- Высота и ширина зависят от содержимого
- Следуя друг за дружкой, инлайновые элементы будут идти в одной строке, не вызывая переноса
- Можно задавать только горизонтальные отступы
- К строчным элементам относятся такие теги как: `<a>`, ``, ``, ``

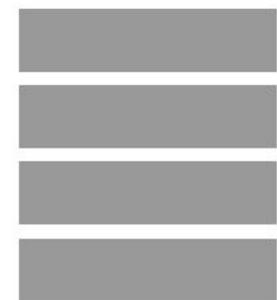
Блочно-строчные элементы



Блочно-строчные элементы

- Имеют форму прямоугольника
- Размеры определяются содержимым
- Размеры можно менять (width, height)
- Рисуются в той точке, в которой закончилась отрисовка предыдущего элемента, и не вызывает переноса за собой

BLOCK:



INLINE:



▼ Блочная модель документа

● `display: block;`

● `display: inline;`

● `display: inline-block;`

BLOCK:



INLINE:



Итого

Свойства блочной модели:

- **width** - ширина блока
- **height** - высота блока
- **border** - рамка
- **padding** - отступ от границы блока до содержимого
- **margin** - отступ между границами блока и его родителем и соседям

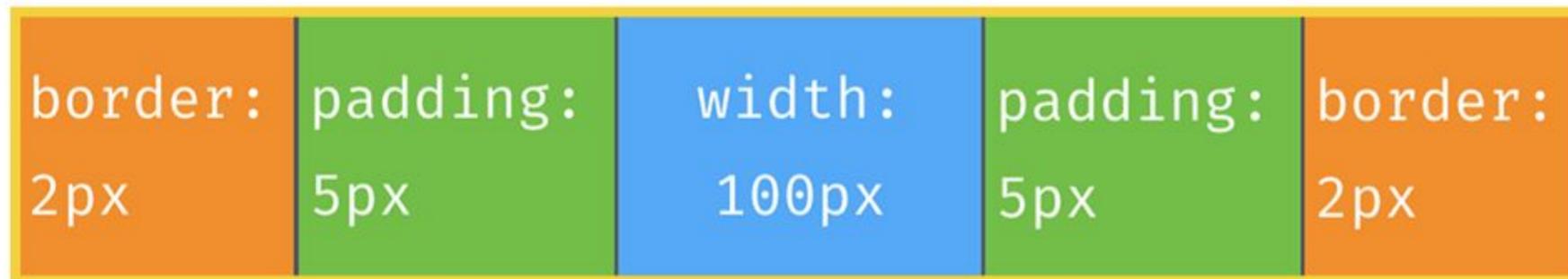
Полный размер блока

- `width` - задает **ширину содержимого**
- **Общая ширина** блока = $\text{width} + 2 * \text{border} + 2 * \text{padding}$
- **margin** не действует ни на общую ширину, ни на ширину содержания

родительский блок

`width: 200px`

`margin:
5px`

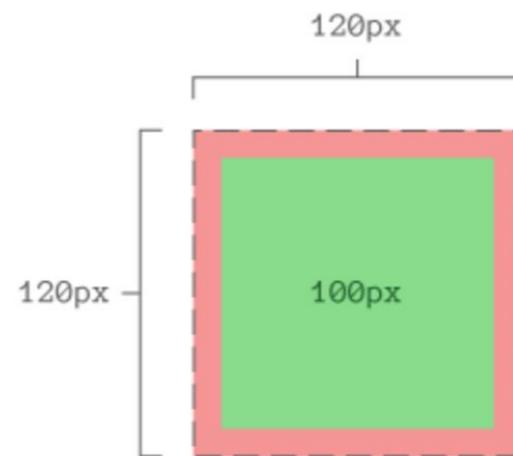


`margin:
5px`

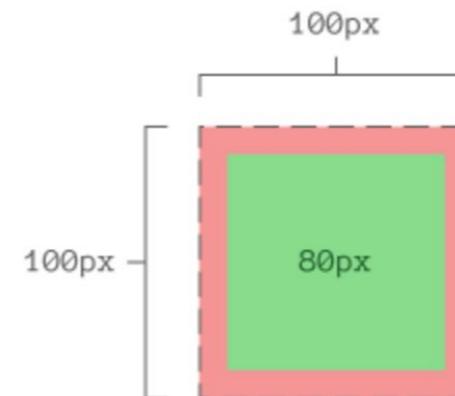
box-sizing: border-box

CSS Box Sizing

content-box **vs** border-box



```
box-sizing: content-box;  
width: 100px;  
height: 100px;  
padding: 10px;
```



```
box-sizing: border-box;  
width: 100px;  
height: 100px;  
padding: 10px;
```



Padding



Content area



Element border

Итого

- Таким образом, итоговое значение ширины и высоты элемента отличается от значений, задаваемых в `width` и `height`
- Мы можем влиять на это поведение через свойство **`box-sizing`**

Оформление текста



font-size

px
%
rem em
vw, vh, vmin, vmax

- Абсолютные: 16**px** (пиксели)
- Относительные: 0.5**em**, 75% (относительно размера шрифта родителя),
rem - относительно размера шрифта в `<html>`
- 1em = длине буквы **M** в данном шрифте
- Словами ***large, small, medium, larger, smaller***

Lorem ipsum dolor sit amet... (*font-size: 10pt*)

Lorem ipsum dolor sit amet... (*font-size: medium*)

Lorem ipsum dolor sit amet... (*font-size: large*)

Lorem ipsum dolor sit amet... (*font-size: 14pt*)

Lorem ipsum dolor sit amet... (*font-size: 150%*)

font-family

- Определяет шрифт и/или семейство шрифтов
- Выглядит так:
font-family: "Times New Roman";
font-family: 'Cuprum', 'Aria', sans-serif;
- Перечисляются шрифты с самого желанного и далее в сторону деградации - в таком порядке их считывает браузер.
- Последним указывают тип шрифта, как самый запасной вариант

font-family: 'Open Sans',
Helvetica, sans-serif

A B C D E F G H I J K L M N

Q R S T U V W X Y Z a b c d

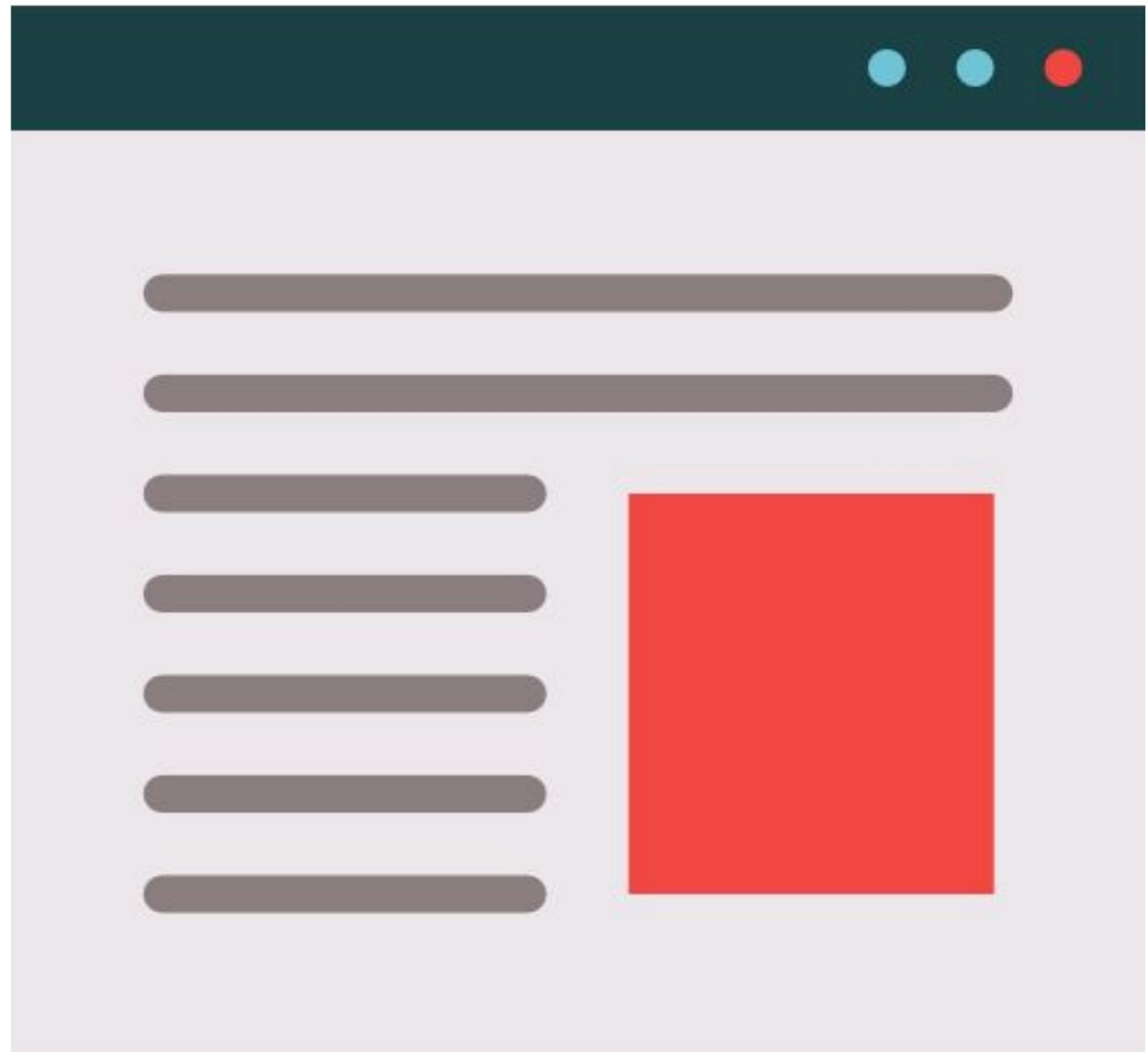
e f g h i j k l m n o p q r s t u v w x

0 1 2 3 4 5 6 7 8 9

! " # \$ % & ' () * + , - . / : ;

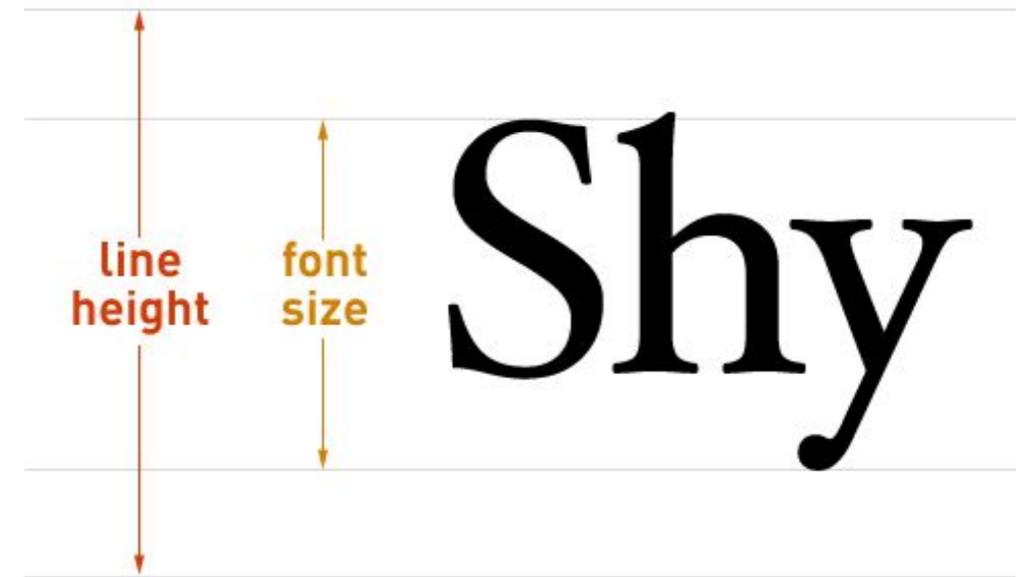
Оформление текста

- font-size
- font-family
- line-height
- font-weight
- font-style
- color



line-height

- Межстрочный интервал/высота строки
- **line-height: 3.5;** - множитель
- **line-height: normal;** - автоматический расчет
- **line-height: 150%;**
- **line-height: 2em;**



Прочее

- font-weight - насыщенность (жирность) шрифта: **bold/normal/bolder/lighter**

- font-style - начертание: **normal / italic**

- **color: #ab12ff**

Демонстрация шрифта - Roboto (font-weight: 100)

Демонстрация шрифта - Roboto (font-weight: 100;font-style: italic)

Демонстрация шрифта - Roboto (font-weight: 200)

Демонстрация шрифта - Roboto (font-weight: 200;font-style: italic)

Демонстрация шрифта - Roboto (font-weight: normal)

Демонстрация шрифта - Roboto (font-weight: normal;font-style: italic)

Демонстрация шрифта - Roboto (font-weight: 600)

Демонстрация шрифта - Roboto (font-weight: 600;font-style: italic)

Демонстрация шрифта - Roboto (font-weight: bold)

Еще свойства для текста:

- `text-transform` - регистр
- `text-decoration` - подчеркивание/перечеркивание
- `text-shadow` - тень
- `white-space` - управляет пробелами
- `text-align` - выравнивание по горизонтали
- `vertical-align` - выравнивание по вертикали

Подведем итог.

Спасибо за
внимание!

CSS

IS

AWESOME

Дополнение

Значения

CSS-правило

```
.main-content h1 {  
  color: #ace503;  
  font-size: 20px;  
  position: relative;  
}
```

Селектор, далее в фигурных скобках правила
в виде пар ключ: значение

Какие бывают значения?

- **Абсолютные** - твердые

```
.adv-text {  
    font-size: 18px;  
}
```

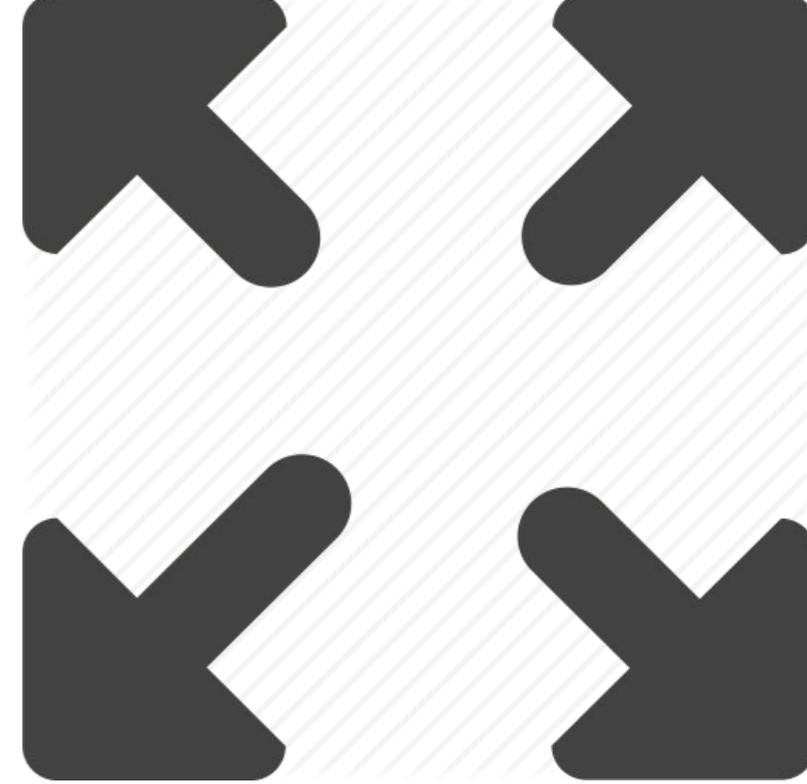
- **Относительные** - зависят от других значений. Для разных типов элементов правила вычисления % различны.

```
.adv-text-container {  
    width: 65%;  
    height: 100vh;  
}
```

- При отрисовке страницы все относительные значения становятся абсолютными

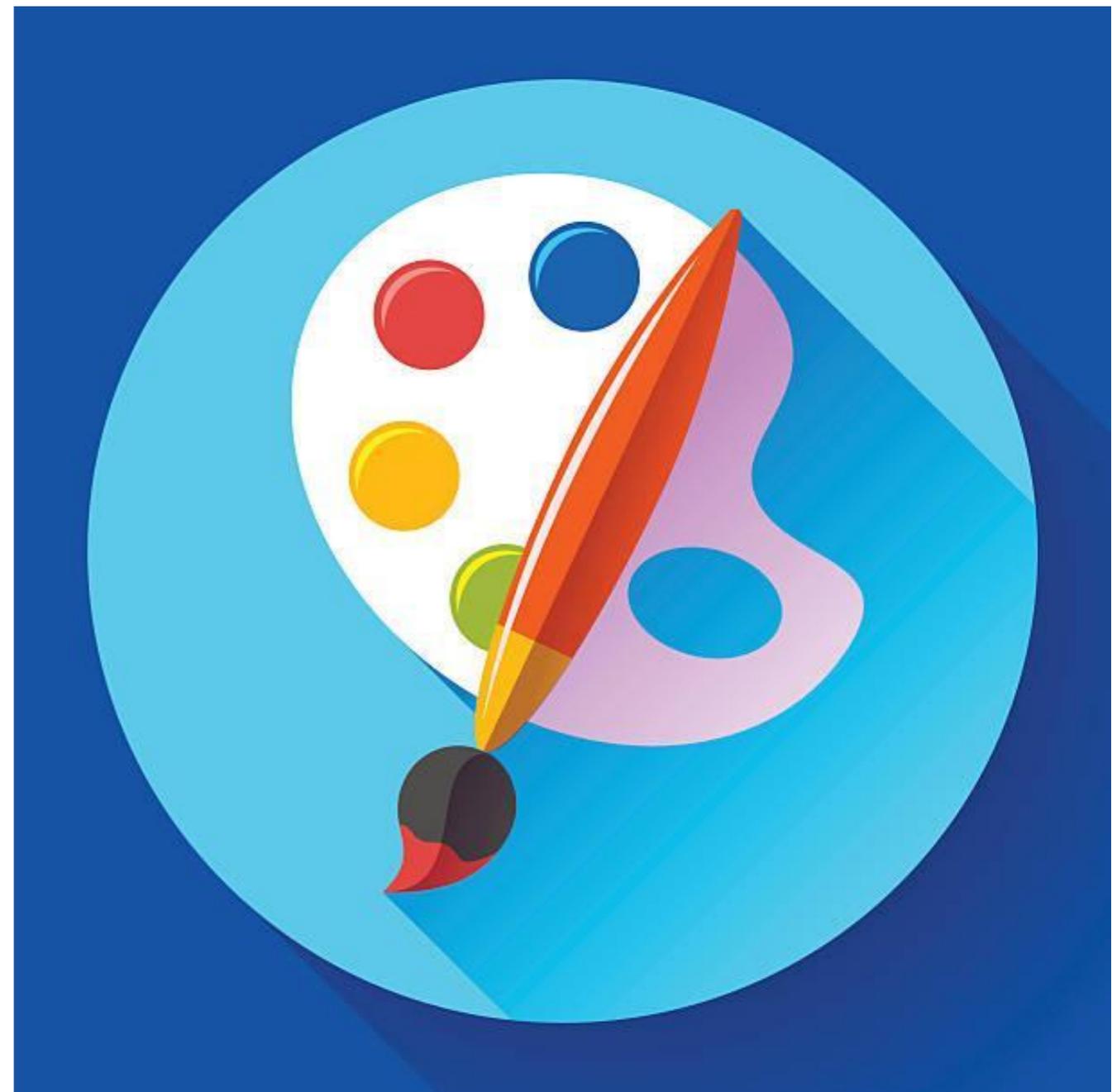
Размеры

- **Пиксели** 20px
- **em** - Эм; - значение в пикселях вычисляется от размера шрифта родителя
- **rem** - значение в пикселях вычисляется от размера шрифта корневого элемента (тега html)
- **%** - как правило, значение вычисляется от аналогичного значения родителя - но бывает иначе
- **vh, vw** - от размеров области просмотра (viewport)



Цвета color

- СЛОВАМИ: **red, lightcoral, green**
- **rgb**: #98dd43,
rgb(127,127,127),
rgba(255, 255, 0. 0.5)
- **hsl**: hsl(159, 21%, 38%)



Ключевые слова

- display: block;
- vertical-align: top;
- color: red;
- ~~text-decoration: line through;~~
- TEXT-TANSFORM: UPPERCASE;

Что, еще?

- **Функции**

width: calc(100%-10px);

- **Текст**

font-family: «Times New Roman»

- Есть еще, но об этом вы узнаете в следующих сериях

```
.main-content h1 {  
    color: #ace503;  
    font-size: 20px;  
    position: relative;  
}
```