



Working with text in HTML and CSS

:: and “pseudo” world



Pseudo-classes

Псевдоклассы - модификаторы селекторов, которые позволяют стилизовать выбранные элементы (например ссылку - 'a') в определенном состоянии (например, если она посещенная - ':visited')

Список всех псевдоклассов можно посмотреть на MDN, рассмотрим часто используемые



Link pseudo-classes

:link — :visited — :hover — :active

При определении состояний ссылок именно такой порядок и важен

a:link - обычная не посещенная ссылка

a:visited - ссылка, по которой пользователь уже переходил

a:hover - элемент, при наведении на него мышкой

a:active - элемент в момент клика на нем



Positional pseudo-classes

:nth-child(n) - указанные потомки / n: even, odd, functional expression, number

:last-child - последний потомок родителя

:first-child - первый потомок родителя

Есть и другие, посмотрите дома

**Еще есть псевдоклассы, которые касаются форм и их элементов, более подробно рассмотрим, после прохождения темы “Формы”.*



Pseudo-elements

Псевдо-элементы имеют похожую идею с псевдоклассами, это модификаторы селекторов, которые позволяют стилизовать некоторые части выбранных элементов (т.е. псевдоклассы - стилизуем состояния, псевдоэлементы - части элементов)

Чаще всего используются `::before`, `::after`

Есть и другие, посмотрите дома



Styling text. HTML

``, `` - полужирный

``, `<i>` - курсив

`<u>`, `<s>` - подчеркнутый и зачеркнутый

`<small>`, `<big>` - маленький и большой шрифт

`<blockquote>` - Цитата

`<pre>` - отображение текста точно так же, как в HTML документе



Styling text. HTML

Стилизация с помощью html тегов - плохая практика. Многие из них морально устарели, другие используются исключительно в семантических целях. Чтобы придать элементам нужный вид - работаем с css

Обратите внимание, что наследование свойств дочерними элементами чаще всего будет касаться тех свойств, которые мы рассмотрим ниже (чаще всего наследование работает со свойствами, влияющими на текст)



Styling text. CSS

`color: #000000;` - цвет шрифта (еще есть `rgb`, `rgba`, и т.д)

`font-size : 16px;` - размер шрифта

`text-align: center;` - выравнивание текста по центру

`font-family: Roboto;` - установка шрифта

`background-color: red;` - цвет фона за текстом

`text-shadow: 2px 2px 1px green;` - тень (смещение по оси X, по оси Y, степень размытия, цвет)



Styling text. CSS

`text-decoration: none;` - отсутствие оформления текста (например подчеркивание ссылок)

`font-style: italic;` - стили шрифта (`normal`, `italic`, *oblique*)

`font-weight: bold;` - толщина шрифта (`normal`, `bold`, `bolder`, `lighter`)

`text-transform: capitalize` - форматирование текста (в данном примере слова начинаются с больших букв)



Fonts

Как уже упоминалось, мы можем изменить шрифт текста с помощью `font-family`.

Существуют “безопасные” шрифты, которые точно будут отображены браузером, так как было задумано

Можно подключить пользовательские шрифты:

- Через внешние сервисы (чаще всего - google fonts)

- Скачать, поместить в папку проекта и подключить в `css`:

```
@font-face {
```

```
font-family:Название;
```

```
src: url("path/to/font");
```

```
}
```



Css units

В css используются два вида единиц измерения - относительные (Relative Lengths) и абсолютные (Absolute Lengths)

Относительные:

em - текущий размер шрифта в браузере, зависит от родителя

rem - размер шрифта браузера, не зависит от родителя, зависит от размера шрифта корневого элемента

% - процент от шрифта родителя

ex - высота символа x



Css units

В css используются два вида единиц измерения - относительные (Relative Lengths) и абсолютные (Absolute Lengths)

Абсолютные:

cm - сантиметр

mm - миллиметр

px - пиксель

in - дюйм