

Основы работы с языком CSS

Мы с вами уже разобрали основные теги языка HTML и теперь пришло время навести красоту. Хотелось бы иметь возможность менять цвет текста, его размер, фон и многое другое. Это делается с помощью языка CSS.

Как правило, CSS команды (*стили*) хранятся в отдельном файле, который подключается специальным тегом ко всем HTML страницам нашего сайта.

Преимущество такого подхода в том, что CSS файл один, а HTML файлов может быть любое количество, хоть тысяча. Если мы сделаем изменение в одном месте CSS файла, например, покрасим все абзацы в красный цвет, то эти изменения применятся на всей тысяче HTML страниц, к которым подключен наш CSS файл. Очень удобно и быстро.

Файл со стилями должен иметь расширение `.css`. Чтобы подключить такой файл к HTML странице, в теге `head` следует написать такую конструкцию:

```
1 | <link rel="stylesheet" href="имяФайла.css">
```

В следующем примере к нашему HTML файлу подключается CSS файл `styles.css`:

```
1 | <!DOCTYPE html>
2 | <html>
3 |   <head>
4 |     <meta charset="utf-8">
5 |     <title>Это заголовок тайтл</title>
6 |     <link rel="stylesheet" href="styles.css">
7 |   </head>
8 |   <body>
9 |     <p>
10 |       Это абзац с текстом.
11 |     </p>
12 |   </body>
13 | </html>
```

Задача 1

#mk.Pm.CL.Inr.1

Создайте и подключите ко всем вашим страницам файл `styles.css`.

Как работать с CSS

Каждому тегу в HTML соответствует так называемый *селектор* CSS. К примеру, тегу `<p>` соответствует CSS селектор `p`, с помощью которого мы можем обратиться ко всем абзацам HTML страницы и, например, покрасить их всех одновременно в красный цвет.

После селектора следует ставить фигурные скобки, внутри которых следует писать CSS *свойства*. Свойства и задают цвет, размер шрифта и другие интересные вещи. Их следует писать в таком формате: имя свойства, потом двоеточие, потом значение этого свойства (например, свойство - это цвет, а "красный" - это значение). Потом нужно поставить точку с запятой и можно писать следующее свойство.

Давайте, например, покрасим все абзацы в красный цвет:

```
1 | p {
2 |   color: red;
3 | }
```

Задача 2

#mk.Pm.CL.Inr.2

В вашем файле `styles.css` разместите код, красящий абзацы в красный цвет. Откройте страницы вашего сайта в браузере и убедитесь в том, что все абзацы стали красными.

Другие значения для цвета

Помимо ключевого слова *red*, задающего красный, можно использовать и другие английские слова для цвета, например, *green* - зеленый, *blue* - голубой, *yellow* - желтый, *orange* - оранжевый, *black* - черный, *white* - белый.

Задача 3

#mk.Pm.CL.Inr.3

Используя соответствующие селекторы покрасьте заголовки `h1` в зеленый цвет, заголовки `h2` в голубой, заголовки `h3` - в красный, а абзацы - в оранжевый.

Способы задания цвета на CSS

В предыдущем уроке вы увидели, как можно задать цвет ключевым словом. Ключевых слов, однако, немного, и с их помощью нельзя сделать все возможные оттенки всех цветов. Для этого существует еще два способа задания цветов. Через `rgb` и через решетку `#`.

Смотрите пример задания цвета через `rgb`:

```
1 | p {  
2 |   color: rgb(255, 236, 114);  
3 | }
```

Смотрите пример задания цвета через `#`:

```
1 | p {  
2 |   color: #ff0000;  
3 | }
```

Для того, чтобы получить нужный цвет, используются специальные программы. Вот пример онлайн генератора цвета: colorscheme.ru, а вот пример программы, позволяющей замерять цвет: colorcop.net.

#mk.Pm.CL.Cl.1

Задача 1

С помощью описанных программ сгенерируйте цвет в одном и во втором формате и примените эти цвета к каким-нибудь тегам страницы.

Ширина и высота элементов на CSS

Для задания ширины и высоты элементов в CSS предназначены свойства `width` и `height`. Значения этих свойств, как правило, задаются в пикселях. Пиксели обозначаются как `px` и добавляются к числу. Например, чтобы задать размер в 100 пикселей, значением свойства следует указать следующее: `100px`.

Посмотрим на работу этих свойств на примере. Зададим ячейкам какой-нибудь таблицы ширину в `200px` и высоту `100px`:

```
1 <table border="1">
2   <tr>
3     <td>1</td>
4     <td>2</td>
5     <td>3</td>
6   </tr>
7   <tr>
8     <td>4</td>
9     <td>5</td>
10    <td>6</td>
11  </tr>
12  <tr>
13    <td>7</td>
14    <td>8</td>
15    <td>9</td>
16  </tr>
17 </table>
```

```
1 td {
2   width: 200px;
3   height: 100px;
4 }
```

Результат выполнения кода:

1	2	3
4	5	6
7	8	9

Задача 1

#mk.Pm.CL.WH.1

Сделайте в HTML коде абзацы с длинным текстом. Средствами CSS поставьте всем абзацам ширину `300px`.

Задача 2

#mk.Pm.CL.WH.2

Дана следующая таблица:

```
1 <table border="1">
2   <tr>
3     <td>Иван</td>
4     <td>Иванов</td>
5     <td>200$</td>
6   </tr>
7   <tr>
8     <td>Николай</td>
9     <td>Петров</td>
10    <td>300$</td>
11  </tr>
12  <tr>
13    <td>Владимир</td>
14    <td>Сидоров</td>
15    <td>400$</td>
16  </tr>
17 </table>
```

Установите ячейкам этой таблицы ширину в `400px` и высоту в `300px`.

Выравнивание текста на CSS

Сейчас мы с вами научимся выравнивать текст. Для этого используется свойство `text-align`. Текст можно выравнивать по левому краю (значение *left*), по правому (значение *right*), по центру (значение *center*) и одновременно и по правому, и по левому краю (значение *justify*).

Давайте посмотрим на примерах.

Значение right

Выровняем текст в абзацах по правому краю:

```
1 <p>
2   абзац с текстом
3 </p>
```

```
1 p {
2   text-align: right;
3 }
```

Результат выполнения кода:

абзац с текстом

Значение center

А теперь выровняем по центру:

```
1 <p>
2   абзац с текстом
3 </p>
```

```
1 p {
2   text-align: center;
3 }
```

Результат выполнения кода:

абзац с текстом

Значение left

А вот значение `left` для абзацев можно и не ставить - они по умолчанию и так выровнены по левому краю. Однако есть элементы, которые по умолчанию стоят по центру (это, например, теги `th`, которые делают ячейку-заголовок таблицы). И вот для них как раз-таки и может потребоваться выравнивание по левой стороне. Сделаем это:

```
1 <table border="1">
2   <tr>
3     <th>Имя</th>
4     <th>Фамилия</th>
5     <th>Зарплата</th>
6   </tr>
7   <tr>
8     <td>Иван</td>
9     <td>Иванов</td>
10    <td>200$</td>
11  </tr>
12  <tr>
13    <td>Николай</td>
14    <td>Петров</td>
15    <td>300$</td>
16  </tr>
17 </table>
```

```
1 th {
2   text-align: left;
3 }
4
5 table {
6   width: 400px;
7 }
```

Результат выполнения кода:

Имя	Фамилия	Зарплата
Иван	Иванов	200\$
Николай	Петров	300\$

Значение justify

Ну, и наконец выровняем текст одновременно и по правому, и по левому краю. Этот эффект можно увидеть только при наличии достаточно длинного текста:

```
1 <p>
2   длинный текст...
3 </p>
```

```
1 p {
2   text-align: justify;
3   width: 300px;
4 }
```

Результат выполнения кода:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean a dapibus magna, ac interdum nisl. Suspendisse eget il fringilla nibh, eu commodo arcu. Donec lacinia tempor velite sed tincidunt. Aliquam porttitor nulla purus, vel vulputate ipsum faucibus sed. Phasellus sodales, lorem vel cursus vehi ante purus lacini dui, interdum fringilla massa eros ut duir.