

Тема урока:

Оформление документа. Таблица каскадных стилей CSS

CSS - (Cascading Style Sheets, каскадные таблицы стилей)

HTML — лишь первый этап в процессе обучения созданию сайтов. Следующим шагом является изучение стилей или CSS. Стили представляют собой набор параметров, управляющих видом и положением элементов веб-страницы. Чтобы стало понятно, о чем идет речь, посмотрим на рис. 1.1.

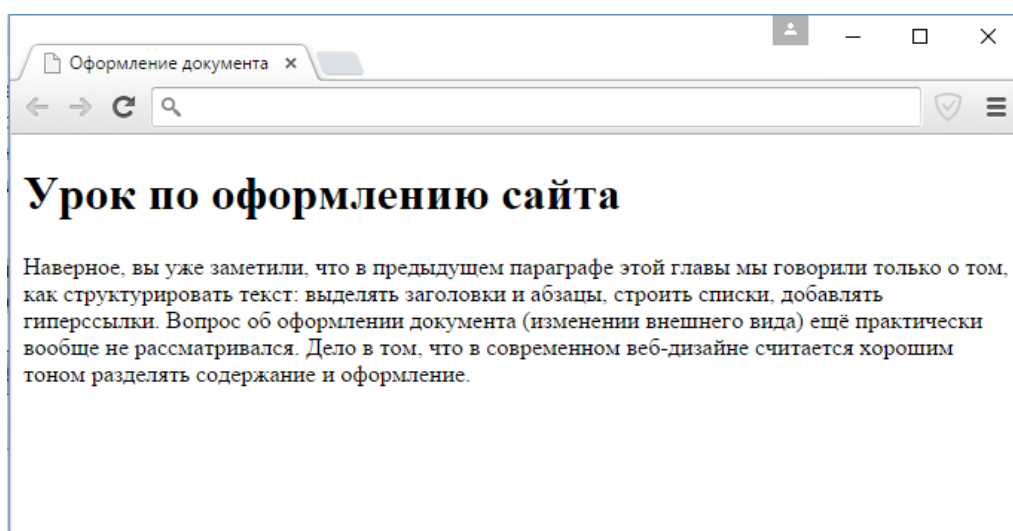


Рис. 1.1. Веб-страница, созданная только на HTML (пример 1)

Это обычная веб-страница, оформленная без всяких изысков. Тот же самый документ, но уже с добавлением стилей приобретает совершенно иной вид (рис. 1.2).

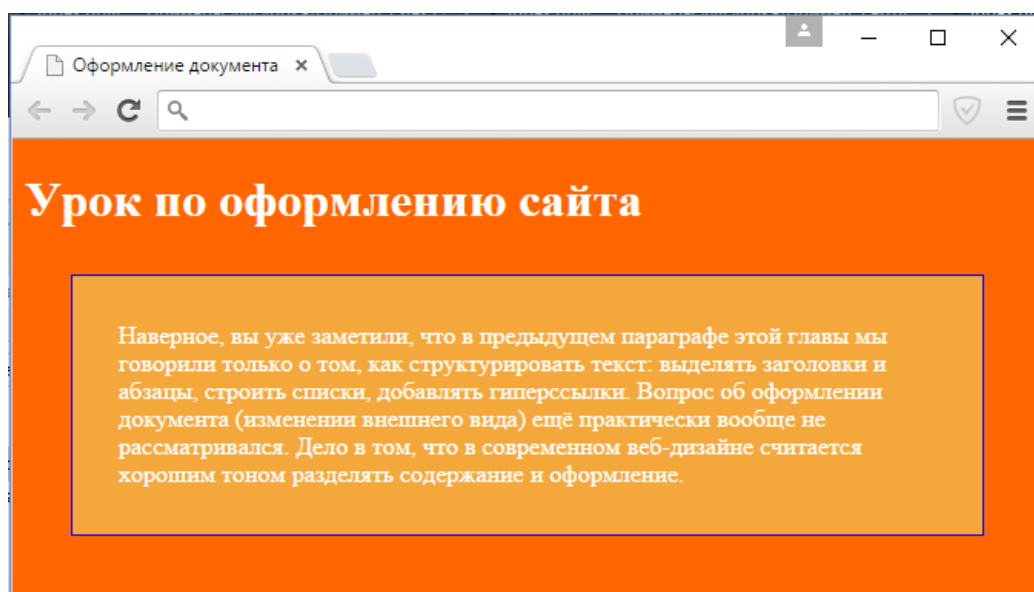


Рис. 1.2. Веб-страница, созданная на HTML и CSS (пример 2)

Как правило, для всего сайта используется 1 файл стиля. Этот файл не напоминает HTML, они используют свой формат. Стилиевой файл - это простой текстовый файл в котором задаются свойства тегов. Обычно стилевые файлы имеют расширение ***.css**

Пример: Чтобы добиться результата, как на рисунке 1.2, нужно создать стилевой файл (например test.css) и записать в него следующее:

```
body {
    color: white;
    background: #ff6600;
}
p {
    border: 1px solid blue;
    background: #F4A83B;
    margin: 30px;
    padding: 30px;
}
```

Сначала указывается селектор. Затем в фигурных скобках определяются свойства.

Определение. Селектор (англ. select - выбирать) название элемента или обозначение для группы элементов, оформление которых меняется.

В данном случае в теге <body> изменяется свойство color (от англ. цвет) и background (от англ. фон). Значение свойства указывается через двоеточие. В конце каждого описания ставится знак ";".

CSS предлагает нам четыре удобных способа указать цвет.

1. По имени. Используются англоязычные названия цветов, например blue или red (см. Именованные цвета).
2. В шестнадцатичном RGB (например, #71C354 или #3C9). Этот способ широко используется в HTML.
3. В десятичном формате RGB (например, rgb (126, 6, 9)), где значение каждой компоненты цвета может варьироваться от 0 до 255.
4. В формате RGB с указанием величин в процентах, например, rgb (30%, 40%, 70%), где значение каждой компоненты цвета может варьироваться от 0 до 100.

В первом случае можно просто написать цвет на английском языке, в остальных цвет указывается в зависимости от того, в какой позиции в нем смешаны красный, зеленый и синий цвета.

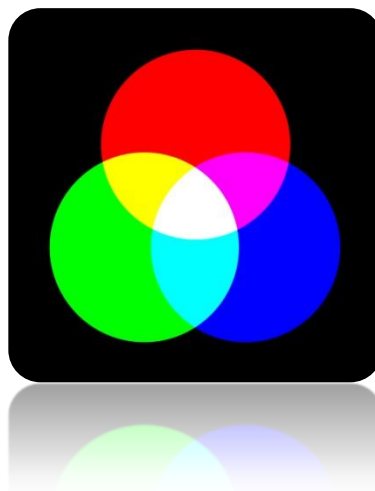


Рис 1.3 Пример смешивания цвета в модели RGB

В нашем случае `color:white;` - обозначает цвет текста белый, `background: #ff6600;` обозначает оранжевый цвет. Есть специальные таблицы, где описаны цвета и их 16-чные значения. Такие таблицы можно найти в интернете, так же можно узнать эти значения в специальных программах (например Photoshop или Gimp)

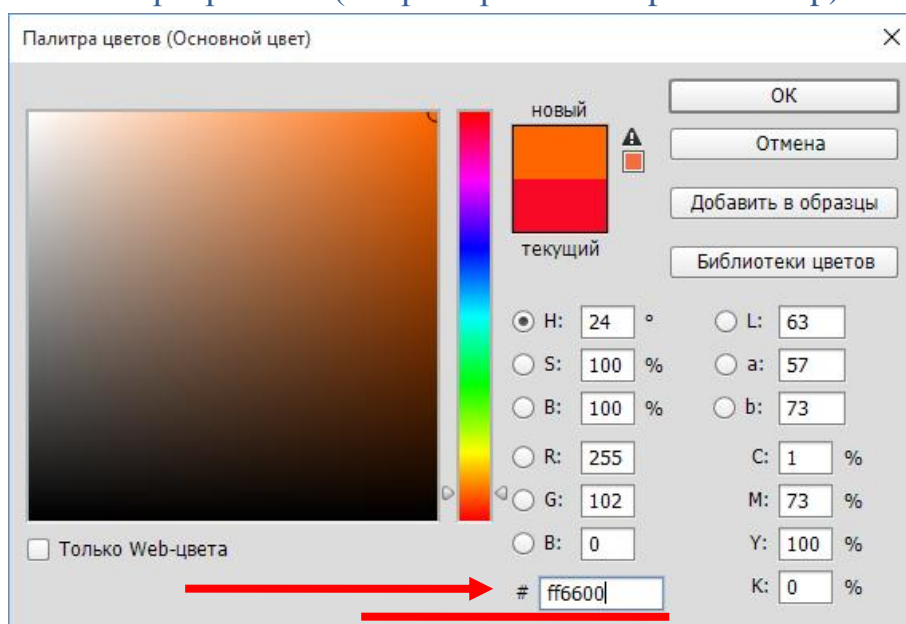


Рис 1.4 Пример получения цвета в Photoshop

После того, как стилевой файл был подготовлен, его нужно подключить к документу HTML. Для этого используется специальный тег `<link>`. Данный тег нужно расположить внутри контейнера `head`.

```
<link rel="stylesheet" type="text/css" href="test.css">
```

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="test.css">
    <meta charset="utf-8">
    <title>Оформление документа</title>
  </head>
```

В этом теге есть 3 атрибута (свойства)

1. **rel** (от англ. relation отношение) определяет, какую роль играет файл test.css, в данном случае stylesheet - таблица стилей.
2. **type** - тип данных, значение text/css говорит, что это файл с каскадной таблицей стилей.
3. **href** - ссылка на файл стилей.

Наследование

Задав стиль для body мы определили свойства по умолчанию для всех вложенных в него элементов color (цвет текста) передалось для абзацев, заголовков и т.д. Это свойство унаследовалось.

Определение. Наследование Наследованием называется перенос правил форматирования для элементов, находящихся внутри других. Такие элементы являются дочерними, и они наследуют **некоторые** стилевые свойства своих родителей, внутри которых располагаются.

Свойство background не наследовалось и вложенные элементы останутся с прозрачным фоном.

Итак можно задавать оформление разным элементам. Разберем теперь свойства для абзаца, на рисунке 1.2.

1. border: 1px solid blue;
2. background: #F4A83B;
3. margin: 30px;
4. padding: 30px;

1. Свойство border - граница абзаца, здесь через пробел указаны 3 параметра

- 1px - это толщина границы в 1 пиксель;
- solid - тип линии сплошная;

- blue - цвет границы.
2. background - указан фон отдельно для абзаца.
 3. margin - поля - 30 пискелей см. Рис 1.5
 4. padding - отступ 30 пикселей. см. Рис 1.5

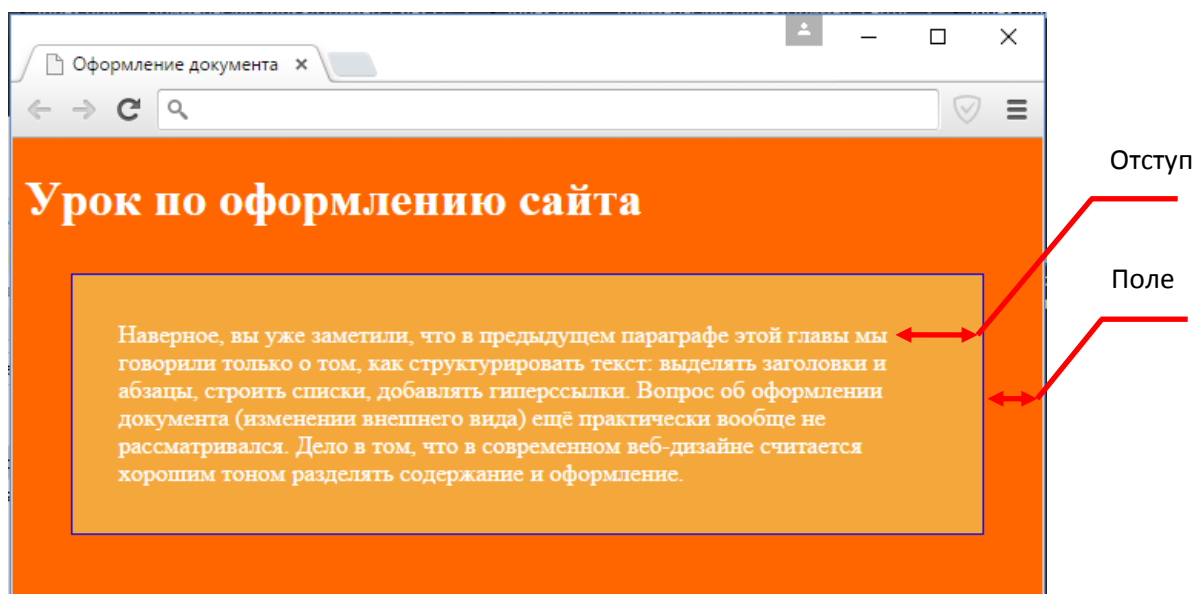


Рис 1.5 Пример полей и отступов

Добавим фон для абзаца и фоновое изображение отдельно для абзаца, для этого добавим свойство background к тегу p в файле стиля.

```
background: top left url(image2.png) no-repeat #E05A6A;
```

здесь все параметры перечислены через пробел. top - означает что изображение будет вверху абзаца left - значит, что оно будет слева no-repeat - что оно не будет повторяться, а будет выведено 1 раз (по умолчанию оно выводится много раз) url(image2.png) - это путь к изображению (в данном случае оно должно храниться в той же папке, где и файл стиля). #E05A6A - заливает остальной фон абзаца. Получим страницу см. рис 1.6.

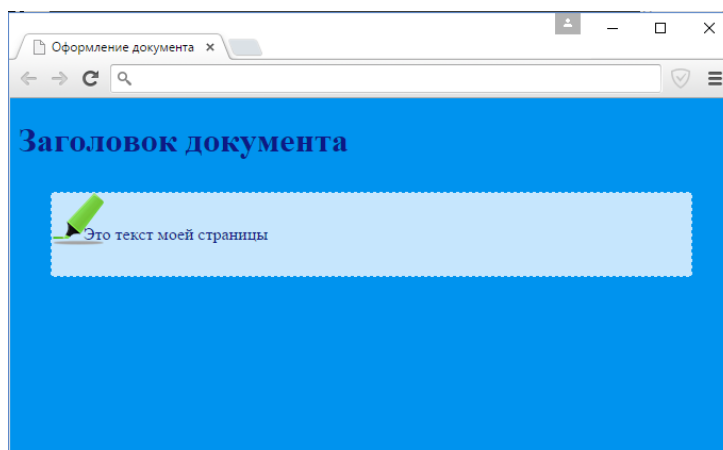


Рис 1.5 Пример абзаца с изображением (пример 2)

Настройка гиперссылок

Для настройки цвета гиперссылок измените стиль для тега `<a>`

`a { color:green; }` - вид ссылки [Ссылка на объект](#)

Определение. **Псевдокласс** определяют динамическое состояние элементов, которое изменяется с помощью действий пользователя.

Пример псевдокласса

- `a:hover` срабатывает при наведении мыши на ссылку.
- `a:visited` - для посещенных страниц (если на них уже нажимали)

`a:hover { color: red; }` вид ссылки, когда на нее наведут курсор мыши [Ссылка на объект](#) (пример со ссылками)

Классы

В рассмотренных примерах оформление определялось для всех одноименных тегов на странице. Если добавить абзац, он будет с тем же оформлением. Часто бывает нужно выделить с помощью специального оформления не все абзацы. В этом случае используются классы. Нужно добавить для абзаца атрибут `class` и указать имя. Например `<p class="error">Ошибка, что-то с памятью</p>`. А в файле стиля нужно описать класс следующим образом: `p.error { ...новое оформление... }`

```
p {
  border: 1px dashed red;
  background: top left url(image2.png) no-repeat #E05A6A;
  margin: 30px;
  padding: 30px; }
```

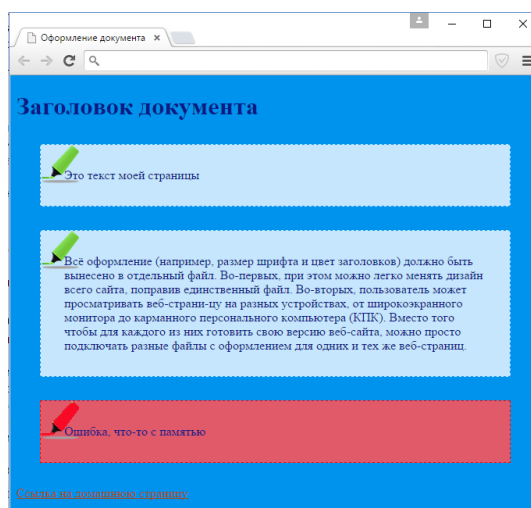


Рис 1.5 Пример абзаца с классом (пример 3)

Стили так же можно записать не в отдельном файле, а прямо внутри контейнера <head>. Для этого используют специальный контейнер <style>.

```
<!DOCTYPE html>
<html>
  <head>
  <style>
  body {
    color: #0E1985;
    background: #0093ee;
  }
  p {
    border: 1px dashed #0495ee;
    background: top left url(image.png) no-repeat #c6e6fd;
    margin: 30px;
    padding: 30px;
  }
  a {
    color: #a94a17;
  }
  a:hover{
    color: #f0c1a7;
  }
  a:visited{
    color: green;
  }
  p.error {
    border: 1px dashed #0495ee;
    background: top left url(image2.png) no-repeat #E05A6A;
    margin: 30px;
    padding: 30px;
  }
</style>
  <meta charset="utf-8">
  <title>Оформление документа</title>
</head> (Пример 4)
```

В комментариях (подписях) к рисункам использовались надписи пример 1 и т.д. В архиве вместе с данным документом есть примеры html страниц и файлов стиля, они находятся в папках с названиями пример 1 и т.д.

Используемая литература:

1. § 26 учебника "Информатика. 11 класс. Углубленный уровень" Авторы учебника — К.Ю. Поляков и Е.А. Еремин;
2. Сайт <http://htmlbook.ru/> Справочник по CSS и HTML
3. Сайт <http://css.manual.ru/> (CSS справочник).,
4. Книга "Изучаем HTML и CSS" Эрика и Элизабет Фримен