**Види стилів.**

* Вбудовані стилі
* Глобальні стилі
* Пов’язані(зовнішні) стилі

**Вбудовані стилі.** Вбудовуються безпосередньо у тег

<h1 style="color: red">Title</h1>

**Глобальні стилі.** Прописуються у парному тегу <style> всередині тега <head>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Document</title>

<style>

h1{

color: red;

text-align: center;

}

</style>

</head>

<body>

<h1>Title</h1>

</body>

</html>

**Пов’язані стилі** – це .css –файли

Що таке css – файл опис зовнішнього вигляду наших сторінок. CSS створюється на базі окремого файлу з розширенням .css Після потрібно застосувати наш css-файл до нашого html-файлу. Це робиться завдяки додавання в *<head>* наступного тегу з атрибутами і посиланням:

*<link rel="stylesheet" href="ім’я css – файлу.css"/> -* тут ми кажем, що посилання до нашого файлу є релевантним і буде використовуватись для стилізування. Синтаксис буде таким і лише таким!

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Document</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<h1>Title</h1>

</body>

</html>

*Тег <link> - є одинарним, його можна закривати, але в html5 це не обов’язково.*

Зберігаємо наш html-файл.

Тепер відкриваєм наш css-файл і робимо першу стилізацію.

Синтаксис у css-файлi наступний:

*Селектор{*

*Властивість стилю: значення ;*

*Властивість стилю: значення;*

*}*

*Приклад:*

body {

background-color: yellow;

color: red;

}

Зберігаємо css-файл і запускаєм наш html-файл

Що вiдбулось? Фон нашої сторінки став жовтим, а текст червоним.

Якщо ні, тоді дивимось:

- чи не допустили ми помилок в підключенні css-файлу?

- чи зробили .css розширення в тегу *<link>?*

*-* чи зберегли ви css та html файли?

**Пріоритетність стилів.**

Якщо стилізувати один параметр одного тегу(на приклад колір фону) за допомогою наступного поєднання різних стилів, ми побачимо наступне:

1.При поєднанні вбудованого(через нього задамо жовтий колір фону) і глобального стилю(через нього задамо червоний колір фону), побачимо, що фон став жовтим. Тобто пріоритетність між двома цими стилями наступна вбудований>глобальний

2.При поєднанні глобального (через нього задамо зелений колір фону) і пов’язаного стилю(через нього задамо чорний колір фону), побачим, що фон став зеленим. Тобто пріоритетність між двома цими стилями наступна глобальний>пов’язаний

Тепер ми маємо наступну пріоритетність: вбудований>глобальний>пов’язаний.

Ще раз про стилі:

1.Вбудовані стилі-пріоритетність № 1. Прописуються в будь який тег за допомогою атрибуту *style=”xxx”*, або через конкретний атрибут (на приклад *bgcolor=”red”)*

2.Глобальні стилі - пріоритет №2. Запис в html документі за допомогою тегу <style></style> в <body> або <head> синтаксис як в css.

3.Пов'язані стилі - пріоритет №3. Робляться допомогою css документу. щоб легше орієнтуватися в коді - стоврюється документ css підключається до html документу через <link rel="stylesheet" href="style.css">

На даний момент ми вже знаємо, що для стилізування у css використовуються селектори.

**Селектори: класи та ідентифікатори.**

Є три різновиди селекторів:

* Селектор по тегу;

body {

background-color: red;

}

* Селектор по id;
* Селектор по класу

Що ж таке селектори по класу та індентифікатору:

Дуже часто буде потрібно доступатись до окремих тегів нашої структури. Це можна зробити через селектор *id* та *class*.

Синтаксис цих дій буде наступним:

В нас є параграф *<р>*, і нам потрібно за стилізувати його у нашому css-файлі.

Ми можемо зробити це наступним чином:

CSS:

p {

color: red;

}

На цьому етапі в нас зміниться колір тексту нашого параграфу.

А що буде якщо в нашій структурі багато тегів *<p>*?

Тоді при цих діях, в нас зміниться колір тексту в усіх параграфах <p>.

Що треба зробити?

Треба надати параграфу індивідуальне ім’я. Його можна надати завдяки атрибутам *id* або *class.*

<p id="tato">Some text</p>

або

<p class="mama">Some text</p>

Як тепер доступитись до цього ім’я у нашому css-файлі?

Синтаксис буде наступним:

CSS

#tato {

color: red;

}

.mama {

color: yellow;

}

Якщо ви помітили, ми доступаємось до id через ‘#’, а до класу через ‘.’

Зверніть увагу: не можна починати ім’я класу, чи ідентифікатору з цифри, та не можна розділяти слова класу пробілами(чому? Про це трохи пізніше).

Не можна:

<p id=”1 selection”>XXX</p>

<p class=”selection 1”>XXX</p>

Дозволяється:

<p id=”one-selection”></p>

<p class=”selection-1”></p>

Класи можна використовувати багато разів, ідентифікатори – тільки один раз.

**Додавання класів.**

Розглянемо ситуацію, коли ми маємо теги *<p class=”one”></p> та <p class=”two”></p>* , та нам потрібно, щоб в них обох був колір букв синім.

Ми можем пропистати у css для кожного класу *color:blue;* ,а можем створити окремий class, наприклад – *color-of-txt*, і під’єднати його до наших тегів наступним чином

*<p class=”one color-of-txt”></p> та <p class=”two color-of-txt ”></p>*

Якщо ж, обидва класи впливають на один і той саме параметр, тоді пріоритет набуде той клас, котрий описаний в css файлі нижче.

Приклад:

<p class="one color-of-txt">Параграф з поєднаними классами 1</p>

<p class="two color-of-txt"> Параграф з поєднаними классами 2</p>

Css:

.one {

color: red;

}

.two {

color: blue;

}

.color-of-txt {

color: green;

}

Браузер зробить нам усі букви зеленими, навіть за умови того, що ми прописували кожному класу окремий колір.

Додаткові джерела:

<https://webref.ru/course/css-tutorial/style>

<https://webref.ru/html/attr/id>

<https://webref.ru/html/attr/class>