**Каскадные таблицы стилей**

Каскадные (многоуровневые) таблицы стилей - cascading style sheets (CSS) - это мощный стандарт на основе текстового формата, определяющий представление данных в браузере.

Если формат HTML предоставляет информацию о составе документа, то таблицы стилей сообщают, как он должен выглядеть. Таким образом, каскадные таблицы стилей дают возможность хранить содержимое отдельно от его представления.

Стиль включает все типы элементов дизайна: шрифт, фон, текст, цвета ссылок, поля и расположение объектов на странице.

CSS разрабатывались так, чтобы обеспечить больший уровень контроля над размещением текста и графики.

CSS предполагает 3 типа таблиц стилей - встроенные, внедренные (внутренние) и связанные (внешние).

Существует три метода для применения таблицы стилей к документу HTML:

* ***Встроенный (Inline).*** Этот метод позволяет взять любой тег HTML и добавить к нему стиль. Использование встроенного метода предоставляет максимальный контроль над всеми свойствами web-страницы. Предположим, надо задать внешний вид отдельного абзаца. Можно просто добавить атрибут style к тегу абзаца, и браузер отобразит этот абзац с помощью параметров стиля, добавленного в код.
* ***Внедренный (Embedded).*** Внедрение позволяет контролировать всю страницу HTML. При использовании тега <style>, помещенного внутри раздела <head> страницы HTML, в код вставляются детализированные атрибуты стиля, которые будут применяться ко всей странице.
* ***Связанный (Linked или External).*** Связанная таблица стилей - мощный инструмент, который позволяет создавать образцы стилей *(master styles)*, которые можно затем применять ко всему сайту. Основной документ таблицы стилей (расширение .css) создается web-дизайнером. Этот документ содержит стили, которые будут едиными для всего web-сайта (неважно, содержит одну страницу или тысячи страниц). Любая страница, связанная с этим документом, будет использовать указанные стили.

**Синтаксис таблицы стилей**

Общий синтаксис записи стилей всех элементов HTML выглядит следующим образом:

***элемент (селектор) {атрибут: значение}***

Такую запись называют стилем.

В качестве названий элементов можно использовать имя элемента разметки (тега) на веб-странице. Атрибут определяет свойство отображаемого элемента, а значение – значение этого свойства.

Например, чтобы окрасить все заголовки уровня 1 в белый цвет, нужно написать так:

***h1 {color: white}***

Можно присвоить одно свойство нескольким тегам, просто группируя эти теги в одной записи. В этом примере устанавливается черный цвет для всех трех типов заголовка:

***h1,h2,h3 {color: #000000}***

Так же, как и теги, можно группировать свойства. Для этого достаточно заключить эти определения для свойств в скобки, разделив точками с запятой:

***h2 {color: #000000;font-size: 14pt;font-family: Monaco}***

Этот стиль определяет для заголовков второго уровня черный цвет и шрифт Monaco размером 14 пунктов.

**Наследование свойств тега**

Одной из возможностей, предоставляемых CSS, является возможность одного тега наследовать свойства другого. Это значит, что не надо описывать свойства всех возможных тегов. Например, если пренебречь заданием свойств для тега <***big***>, он унаследует свойства тегов, в которые заключен (в данном случае <***h3***>):

***<h3>Часть 4. <big>Очень большая</big> </h3>***

Другая полезная особенность наследования – то, что оно может быть использовано для применения стилей контекстно. Например, можно не только установить цвет <***h3***> зеленым, а <***big>*** голубым. Можно также задать цвет <***big***> желтым в случае нахождения этой части текста между тегами <***h3***> и ***</h3***> - и только в них. Это очень просто сделать:

***h3 big {color: yellow}***

Здесь таблица стилей определяет, что любое содержание <***big***>, заключенное одновременно в теги <***h3***>, будет показано желтым. Это не сказывается на стиле <***big***> в других местах страницы. Надо только проверить отсутствие запятой в такой конструкции. Иначе этот селектор будет означать, что содержимое обоих тегов будет показано желтым.

Такие правила называются контекстными селекторами, потому что они определяют значения переменных в зависимости от контекста.

Можно определить значения для разных переменных в одной строке, разделив переменные запятой. Например:

***h3 em, h2 i {color: yellow}***

Можно определить несколько правил форматирования для одного и того же тега и с помощью параметра ***class*** применять для соответствующих тегов разные стили. Имя класса указывается в селекторе стиля после имени тега и отделяется от него точкой.

Например, для тега р можно задать разные стили:

***p.t1 {color:blue}***

***p.tc {text-align:center}***  
 и применять их к разным абзацам:

***<p class=”t1”>***

***<p.class=”tc”>.***