

И.В. Смагина, О. А. Романова

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА ПРИЕМОМ РАБОТЫ С HTML И CSS ПРИ СОЗДАНИИ ВЕБ-ДОКУМЕНТОВ

В современной действительности существует много способов создания сайтов, состоящих из веб-страниц. Каждый из них имеет преимущества и недостатки. В статье проведен сравнительный анализ при работе с гипертекстовым языком разметки документов HTML и каскадными таблицами стилей CSS.

Ключевые слова: гипертекстовый язык разметки документов HTML, каскадные таблицы стилей CSS.

UDC 004.91

I.V. Smagina, O.A. Romanova

COMPARATIVE CHARACTERISTIC OF WORKING METHODS WITH HTML AND CSS AT CREATION OF WEB DOCUMENTS

Nowadays there are many ways of sites creation consisting of web pages. Each of them has advantages and disadvantages. In the article comparative analysis is carried out while working with hypertext marking language of documents HTML and cascade tables of CSS styles.

Keywords: hypertext marking language of documents HTML, cascade tables of CSS styles.

HTML (от англ. HyperText Markup Language — «язык разметки гипертекста») — стандартный язык разметки документов во Всемирной паутине. Большинство веб-страниц создаются при помощи языка HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается в виде документа в удобной для человека форме.

Относительная простота освоения и использования HTML привела к тому, что HTML стал базовым для бурно развивающейся индустрии Web. При этом HTML прекрасно подходил под требования раннего периода развития Web, в котором преобладали текстовые документы, снабженные заголовками, списками и гиперсвязями, указывающими на другие документы, однако, как только у Web-авторов возникла необходимость в средствах компоновки страниц и мультимедиа, у языка HTML появились серьезные проблемы, связанные с дальнейшим его развитием, поэтому HTML был расширен во многих направлениях.

HTML дает авторам следующие возможности:

- издавать сетевые документы с заголовками, текстом, таблицами, списками, фотографиями и изображениями, и т.п.;
- получать информацию из Сети через ссылки гипертекста по нажатию кнопки;
- создавать формы для отправки запросов на удаленные компьютеры, чтобы производить поиск информации, осуществлять бронирование, заказывать товары и т.п.;
- включать электронные таблицы, видео клипы, аудио клипы, и другие программные приложения непосредственно в их документы.

Каскадные таблицы стилей (CSS, англ. Cascading Style Sheets) есть один из способов форматирования HTML документа. Тем не менее, философия этого способа радикально отличается от типично разметочного. Недостатки подхода разметки в HTML видны сразу – при попытке сменить оформление сайта приходится менять все страницы и все элементы, один за другим. Таблицы CSS предлагают логический способ оформления документа. Т.е. в любом правильно составленном тексте можно отделить такие понятия, как заголовки, параграфы, термины, ссылки друг от друга. К тому же CSS охватывает и такие области, как оформление изображений, таблиц и прочих визуальных элементов. Сейчас можно с уверенностью сказать, что CSS предоставляет гораздо более широкие возможности для оформления, нежели стандарт HTML.

Итак, CSS – это язык стилей, определяющий отображение HTML-документов. Например, CSS работает с шрифтами, цветом, полями, строками, высотой, шириной, фоновыми изображениями, позиционированием элементов и многими другими вещами. HTML может использоваться для оформления web-сайтов. Но CSS предоставляет большие возможности и более точен и проработан. CSS, на сегодняшний день, поддерживается всеми браузерами (программами просмотра).

Рассмотрим, в чём разница между CSS и HTML. HTML используется для структурирования содержимого страницы. CSS используется для форматирования этого структурированного содержимого. Раньше язык HTML использовался только для вывода структурированного текста. Автор мог только размечать текст: «это – заголовок» или «это – абзац», используя HTML-тэги, такие как <h1> и <p>. По мере развития Web дизайнеры начали искать возможности форматирования он-лайн документов. Чтобы удовлетворить возросшим потребностям потребителей, производители браузеров (тогда - Netscape и Microsoft) изобрели новые HTML-тэги, такие, например, как , которые отличались от оригинальных HTML-тэгов тем, что они определяли внешний вид,

а не структуру. Это также привело к тому, что оригинальные тэги структурирования, такие как <table>, стали всё больше применяться для дизайна страниц вместо структурирования текста. Многие новые тэги дизайна, такие как <blink>, поддерживались только одним из многих браузеров. CSS был создан для исправления этой ситуации путём предоставления web-дизайнерам возможностей точного дизайна, поддерживаемого всеми браузерами. Одновременно произошло разделение представления и содержимого документа, что значительно упростило работу.

Сейчас ситуация с разными браузерами, по-разному понимающими различные атрибуты CSS, понемногу меняется. К примеру, браузеры Internet Explorer и Opera, которыми пользуется подавляющее большинство пользователей, отображают простые таблицы CSS практически одинаково.

В таблице 1 сравним по различным критериям приемы работы с HTML и CSS

Таблица 1. Сравнение HTML и CSS

Критерий сравнения	HTML	CSS
Ключевая лексическая единица	Тэг	Селектор или атрибут style
Приписывание свойств ключевой единице	Для некоторых тегов необязательно	Приписывание свойств обязательно всегда
Основные символы-ограничители	Символы < и > охватывают ключевую единицу	Символы { и } или «и» (в случае атрибута style) охватывают только списки параметров
Дополнительные символы-ограничители		(и) для значения url
Символ присвоения свойства	Знак равенства =	Двоеточие :
Символы-разделители параметров в списках	Пробел	Точка с запятой ;

Таким образом, тэги в HTML записываются внутри веб-документа, например, тэг <P>, отвечающий за разметку параграфов. Каждый тэг имеет набор атрибутов и их возможные значения (для тэга P – это способ горизонтального выравнивания параграфа ALIGN, и значения: left, center, right, justify). И перед каждым абзацем, расположенным на HTML-странице необходимо записать следующее:

```
<P ALIGN= «center»> Текст </P>
```

Рассмотрим способы применения CSS к HTML-документу, т.е. простейший синтаксис CSS и методы подключения таблиц.

Метод 1: Таблицы стилей документа (document style sheets).

Называются так потому, что располагаются непосредственно в HTML-документе и применяются только к нему. Иногда называются embedded style sheet (встроенный стиль). CSS-стили и комментарии располагаются между открывающим и закрывающим тегами элемента style:

```
<style type="text/css">
```

...

```
</style>
```

Сам тег style может находиться в любой части документа, но обычно его размещают внутри элемента head.

Все правила этой таблицы действуют на протяжении всего документа. Например:

```
<html>
  <head>
    <title>Example</title>
    <style type="text/css">
      body {background-color: #FF0000;}
    </style>
  </head>
  <body>
    <p>This is a red page</p>
  </body>
</html>
```

Рисунок 1 - Использование встроенных CSS

Метод 2: Стили, подставляемые в строку (inline styles)

Если нужно назначить стиль отдельному элементу на странице, не применяя внешних стилей и элемента `style`, например, элемент встречается единожды в документе или на сайте, но требует особого оформления, то таблица стилей располагается в теле какого-то отдельного тега (посредством его атрибута `style`) этого документа. Все правила этой таблицы действуют только на содержимое этого тега.

```
<html>
  <head>
    <title>Example</title>
  </head>
  <body style="background-color: #FF0000;">
    <p>This is a red page</p>
  </body>
</html>
```

Рисунок 2 - Стили, подставляемые в строку

Метод 3: Внешняя таблица стилей (файл CSS)

Прикрепляется к HTML-странице посредством написания элемента `link`, который должен располагаться внутри элемента `head` и нигде более.

```
<link rel="stylesheet" type="text/css" href="mystyle.css" media="all" />
```

Встретив в HTML-документе этот тег, браузер загрузит с сайта CSS-файл (в нашем случае это `mystyle.css`) и применит к документу содержащиеся в нём стили. Файл не должен содержать ничего, кроме CSS-инструкций. Внешний файл со стилями удобен тем, что одни и те же стили можно применять ко множеству документов на сайте — в каждом из них достаточно лишь вписать одну строку с элементом `link`

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" media="all"/>
</head>
<body>
<p> This is a red page </p>
</body>
</html>
```

Рисунок 3 - Использование внешних CSS

Итак, оценим преимущества CSS перед HTML. Во-первых, с помощью стилей можно определить вид веб-страницы для разных устройств вывода: монитора, принтера, смартфона, КПК и др. Например, на экране монитора отображать страницу в одном оформлении, а при ее печати — в другом. Эта возможность также позволяет скрывать или показывать некоторые элементы документа при отображении на разных устройствах.

Во-вторых, с помощью стилей можно уменьшить время загрузки страниц сайта за счет переноса правил представления данных в отдельный CSS-файл. В этом случае браузер загружает только структуру документа и данные, хранимые на странице, а представление этих данных загружается браузером только один раз и могут быть закешированы.

В-третьих, с помощью стилей достигается простота последующего изменения дизайна. Не нужно править каждую страницу, а лишь изменить CSS-файл.

В-четвертых, возникают дополнительные возможности оформления. Например, с помощью CSS-вёрстки можно сделать блок текста, который остальной текст будет обтекать (например для меню) или сделать так, чтобы меню было всегда видно при прокрутке страницы.

Но, существуют и недостатки при использовании CSS. Это различное отображение вёрстки в различных браузерах (особенно устаревших), которые по разному интерпретируют одни и те же данные CSS и часто встречающаяся необходимость на практике исправлять не только один CSS-файл, но и теги HTML, которые сложным и ненаглядным способом связаны с селекторами CSS, что иногда сводит на нет простоту применения единых файлов стилей и значительно удлиняет время редактирования и тестирования.

В связи с последним рассмотрим, как поддерживаются CSS различными браузерами. Хотя большинство свойств CSS описаны достаточно давно, разработчики не всегда включали их поддержку в браузер или делали это не в полном объеме. По этой причине часто возникала ситуация, когда стандарты нельзя было применять только потому, что они не работали. К счастью, эта ситуация исправляется и современные браузеры поддерживают спецификацию CSS 2.1 практически в полном объеме, хотя еще и не до конца. Тем не менее, при создании

универсальных веб-страниц, которые корректно отображаются в разных браузерах, необходимо знать, какие свойства будут работать и где, а какие нет. Для этого в таблице приведены популярные браузеры — Internet Explorer, Chrome, Opera, Safari и Firefox. Например, для свойства background-size, которое масштабирует фоновое изображение согласно заданным размерам и имеет синтаксис background-size: [<значение> | <проценты> | auto]{1,2} | cover | contain, таблица выглядит так:

Internet Explorer				Chrome				Opera				Safari				Firefox			
6.0	7.0	8.0	9.0	9.0	10.0	11.0	12.0	9.6	10.0	10.53	11.0	2.0	3.1	4.0	5.0	2.0	3.0	3.6	4.0
✘	✘	✘	✔	✔	✔	✔	✔	✘	✘	✔	✔	✘	✘	✘	✔	✘	✘	✘	✔

А для свойства box-shadow, которое добавляет тень к элементу и имеет синтаксис box-shadow: none | <тень> [, <тень>], *где <тень>:inset <сдвиг по x> <сдвиг по y> <радиус размытия> <растяжение> <цвет >, таблица выглядит так:

Internet Explorer				Chrome				Opera				Safari				Firefox			
6.0	7.0	8.0	9.0	9.0	10.0	11.0	12.0	9.6	10.0	10.50	11.0	2.0	3.1	4.0	5.0	2.0	3.0	3.6	4.0
✘	✘	✘	✔	✔	✔	✔	✔	✘	✘	✔	✔	✘	✘	✘	✔	✘	✘	✘	✔

Итак, из вышеизложенного следует, что при форматировании веб-документов удобнее все-таки применять таблицы CSS, а теги HTML использовать только для разметки страниц. Язык HTML на самом деле достаточно обширный, но на практике приходится использовать порядка 40% всех тегов. Язык HTML и таблицы CSS очень просты и написаны таким образом, чтобы каждый мог легко в них ориентироваться. Сегодня с их помощью любой пользователь может создать свою страничку в Интернете или блог на бесплатных площадках.

Список литературы.

1. Web-технологии: Что такое HTML [Электронный ресурс]/ - Режим доступа: <http://htmlweb.ru/html/bhtml.php/>. Дата обращения 02.03.2012.
2. Каскадные таблицы стилей [Электронный ресурс]. - Режим доступа: <http://ru.wikipedia.org/>. Дата обращения 02.03.2012.
3. Справочник CSS [Электронный ресурс].- Режим доступа: <http://htmlbook.ru/css>. Дата обращения 02.03.2012.

Смагина Ирина Валерьевна

*к.э.н., доцент кафедры информатики и информационных технологий
Орловского государственного института экономики и торговли
e-mail: zsmagin@rambler.ru*

Романова Ольга Алексеевна

*студентка 4 курса финансово-экономического факультета
Орловского государственного института экономики и торговли
e-mail: solnce2476@rambler.ru*