

Динамический HTML и CSS

Цель: познакомить свойствами таблиц стилей и рекомендациями по созданию HTML-документов

Динамический HTML (Dynamic HTML, или DHTML) - это развитие традиционного языка HTML, позволяющее создавать интерактивные, "живые" HTML-документы. Стремлению сделать Web-страницу динамичной и интерактивной, то есть легко изменяющейся, управляемой пользователем, столько же лет, сколько и самому WWW.

Один из методов реализации идеи динамичности заставил разработчиков "дополнить" HTML и встроить в браузеры **средства обработки специальных языков программирования**, таких как JavaScript, JScript и VBScript. Сами сценарии (скрипты), т. е. программы, написанные на одном из этих языков, можно размещать вместе с основным содержанием и разметкой HTML-документа. Программы на этих языках перемещаются по сети Интернет в виде исходных текстов, а компилируются и выполняются после загрузки в конкретный браузер.

Все события, происходящие в браузере, например, нажатие кнопки или переход к другой странице, можно отследить и обработать при помощи программы, написанной на одном из выше указанных языков. Важность этих событий состоит в том, что они позволяют программе и браузеру тесно взаимодействовать друг с другом. Например, программа может определить, когда пользователь покидает страницу, и выполнить при этом необходимые действия.

С противоположного фланга на HTML наступает, принимая на себя часть его функций и снимая тем самым часть его ограничений, **язык иерархических стилевых спецификаций (Cascading Style Sheets, CSS)**. Главная цель CSS (эту технологию часто называют **каскадные таблицы стилей**) - отделить структуру документа от его оформления и позволить автору или пользователю самому решать, как должен выглядеть тот или иной элемент содержания HTML-страницы. Другим достоинством таблиц (иногда называемых листами) стилей является возможность обеспечить единый стиль оформления некоторого набора HTML-документов, например курсовых или дипломных работ.

На стыке этих двух расширений HTML - языка стилей и языка сценариев - и возник тот набор технологий, который сейчас принято называть динамическим HTML. Основную идею динамического HTML можно сформулировать очень просто: полный контроль языка сценариев над всеми без исключения элементами документа, параметрами их оформления и размещения (как подразумеваемыми в HTML, так и задаваемыми с помощью CSS) и даже над самим текстом страницы. Благодаря этому любой элемент HTML-документа может изменять свое содержание и внешний вид.

К сожалению, два варианта реализации динамического HTML, предлагаемые компаниями Netscape Communications Corporation и Microsoft - разработчиками двух наиболее популярных браузеров, имеют между собой очень мало общего, особенно если судить по документации самих этих компаний. Различиям между этими реализациями и рекомендациям по подготовке HTML-страниц, совместимых с разными браузерами, посвящено много статей в Интернет. В этой главе рассматриваются образцы применения DHTML, одинаково отображаемые этими браузерами, и даются советы, как достичь такого "взаимопонимания".

Стили и таблицы стилей

На самом простом уровне, **стиль** является всего лишь правилом, сообщаящим браузеру, как интерпретировать отдельные теги HTML. С каждым тегом связан ряд **свойств**, значения которых определяют его вид при выводе браузером. Так, например, у большинства тегов (<P>, <TD>, и т. д.) есть свойство COLOR, значение которого определяет цвет, используемый для отображения содержимого тега.

Таблица (лист) стилей CSS- набор правил отображения, применяемых в документе, к которому присоединяется соответствующая таблица стилей. Все свойства, используемые в таблицах стилей, условно можно разделить на несколько групп:

- управляющие видом шрифта (гарнитура, кегль, цвет, наклон, жирность,..);
- управляющие форматированием абзаца (отступ красной строки, выравнивание, интерлиньяж, расстояние между словами,..);
- управляющие свойствами блока (отступы слева-сверху-справа-снизу, местоположение блока в документе, видимость блока,..);
- другие (цвет ссылок, изменение внешнего вида курсора,..).

Использование стилей

Одним из способов подключения стиля к HTML-странице (остальные способы рассматриваются в параграфе "Каскадные таблицы стилей") является использование в заголовке документа контейнера `<STYLE>`:

```
<HEAD>
...
<STYLE TYPE="text/css">
...
описание стилей
...
</STYLE>
...
</HEAD>
```

Включение в этот контейнер атрибута `TYPE="text/css"` указывает браузеру, что в нем содержится описание стилей. В настоящее время атрибут `TYPE` может принимать только указанное значение, однако если он будет опущен, то некоторые браузеры могут проигнорировать всю таблицу стилей.

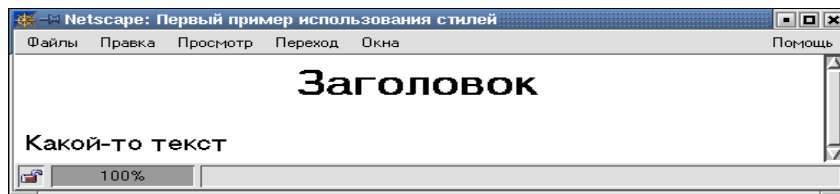
Само описание стиля включает селектор тега, указывающий имя тега, на который распространяется правило стиля, и фигурные скобки, содержащие одну или более пар объявлений вида "свойство_стиля:значение", разделенных точкой с запятой:

```
селектор_тега{
свойство_стиля1:значение1;
свойство_стиля2:значение2;
свойство_стиля3:значение3;
}
```

Если в фигурных скобках указано всего одно правило, то точку с запятой можно опустить.

Пример

```
<HTML>
<HEAD>
<TITLE>Первый пример использования стилей</TITLE>
<STYLE TYPE="text/css">
<!--
H1 {text-align: center}
-->
</STYLE>
</HEAD>
<BODY>
<H1>Заголовок</H1>
<P>
Какой-то текст
</BODY>
</HTML>
```



В данном примере в секции заголовка документа `<HEAD>... </HEAD>` задан стиль (правило), согласно которому браузер должен выровнять по центру страницы содержимое контейнера `<H1>... </H1>`: свойству **text-align** присвоено значение **center**.

Обратите внимание на использование тегов комментария (`<!--` и `-->`). Все содержимое контейнера `<STYLE >` находится между ними. Это позволяет скрыть информацию о стиле от браузеров, которые не распознают тег `<STYLE >` (в противном случае они могут принять правила за текст и вывести их на экран).

В таблицах стилей также можно использовать **комментарии**, позволяющие описать то или иное свойство: текст, расположенный между символами `/*` и `*/` не обрабатывается браузером и **не рассматривается, как элемент таблицы стилей** (не путайте теги комментария и комментарии таблиц стилей, они имеют разный синтаксис и разное назначение).

Пример

Рассмотрим пример оформления текста. Установим белый цвет фона документа (свойство **background-color**), затем задействуем свойства **управления шрифтом**: **color** (цвет), **font-family** (гарнитура шрифта) и **font-size** (размер шрифта), например,

- цвет - серый;
- семейство шрифта (гарнитура) - Times, шрифт с засечками;
- размер шрифта - 14 пунктов.

Добавим свойства, связанные с **форматированием абзаца**: **text-indent** (отступ красной строки), **text-align** (выравнивание), **padding-left** (отступ слева), например,

- дополнительный отступ красной строки - 10pt;
- выравнивание - по ширине документа;
- отступ от левого края - 20pt.

Установим темно-красный цвет (свойство **color**) заголовков с первого по третий уровень.

Документ примет вид:

```
<HTML>
<HEAD>
<TITLE>Второй пример использования стилей</TITLE>
<STYLE TYPE="text/css">
<!--
BODY {background-color: white}

H1, H2, H3 {color: red}

P {
/* шрифт */
  color: gray;
  font-family: Times, serif;
  font-size: 14pt;
/* абзац */
  text-indent: 10pt;
  text-align: justify;
  padding-left: 20pt;
}
```

```
-->  
</STYLE>  
</HEAD>  
<BODY>  
  
...  
</BODY>  
</HTML>
```