

8. Назначение и применение CSS

Цель: изучение CSS.

8.1. Теоретические сведения

8.1.1. Назначение CSS

Спецификация CSS (Cascading Style Sheets) позволяет остаться в рамках декларативного характера разметки страницы и полностью контролировать форму представления элементов HTML-разметки.

Каскадные таблицы стилей призваны разрешить противоречие между точностью определения размеров картинок и приложений, с одной стороны, и точностью определения размеров блоков текста и его начертания — с другой.

CSS позволяет полностью переопределить форму представления элемента разметки по умолчанию.

Практическое значение CSS для Web-инжиниринга (совокупности технологий разработки и сопровождения Web-узлов) заключается в том, что процесс создания узла можно формализовать и представить в виде последовательности действий:

- необходимо определиться с номенклатурой страниц, т. е. все страницы проектируемого Web-узла разбить на типы, например, домашняя страница, навигационные страницы, информационные страницы, коммуникационные страницы и т.п. У каждого узла этот перечень может быть свой;
- для каждого типа страниц требуется разработать определенную логическую структуру (стандартный набор компонентов страницы);
- следует создать навигационную карту узла и форму ее реализации на страницах;
- для каждого стандартного компонента страницы нужно разработать стиль его отображения (CSS-описатель);
- теперь остается только рисовать картинки, создавать анимацию, писать программы, вручную вводить текст и графику или генерировать содержимое страниц автоматически при обращении к ним.

8.1.2. Способы применения CSS

Под способами применения CSS мы в данном разделе понимаем форму декларирования стиля на HTML-странице и форму связывания описания стиля отображения элемента разметки с самим элементом. Речь идет о том, где и в какой форме автор страницы (или дизайнер) описывает стиль, и как и в какой форме на него ссылается.

Итак, различают четыре способа применения стилей:

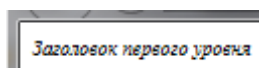
- переопределение стиля в элементе разметки;
- размещение описания стиля в заголовке документа в элементе STYLE;
- размещение ссылки на внешнее описание через элемент LINK;
- импорт описания стиля в документ.

8.1.2.1. Переопределение стиля

```
<H1 STYLE="font-weight:normal; font-style:italic; font-size:10pt;">
```

Заголовок первого уровня

```
</H1>
```



Атрибут style можно применить внутри любого элемента разметки. Например, мы можем через style определить ширину и выравнивание элемента hr (горизонтальное подчеркивание):

```
<HR STYLE="width:100px;">
```

Здесь же нужно отметить следующее: стили разработаны в первую очередь для управления отображением текста. Не следует увлекаться стилями при управлении отображением нетекстовых элементов HTML-разметки.

8.1.2.2. Элемент STYLE

Применение элемента STYLE — это основной способ внедрения каскадных таблиц стилей в ткань HTML-документа. Помимо управления отображением элементов разметки, элемент STYLE позволяет описывать стилевые свойства элементов, которые можно изменять при программировании на JavaScript.

Элемент STYLE дает возможность определить стиль отображения для:

- стандартных элементов HTML-разметки;
- произвольных классов (селектор CLASS);
- HTML-объектов (селектор ID).

8.1.2.3. Ссылка на внешнее описание

Ссылка на описание стиля, расположенное за пределами документа, осуществляется при помощи элемента LINK, который размещают в элементе HEAD. Внешнее описание может представлять собой

файл, содержащий описание стилей. Описание стилей в этом файле будет по синтаксису в точности совпадать с содержанием элемента STYLE.

Ниже приведен пример ссылки на внешнее описание стилей:

```
<LINK TYPE="text/css" REL="stylesheet"
      HREF="http://intuit.ru/my_css.css">

<link type="text/css" rel="stylesheet"
      href="http://www.pac.by/my_css.css">
```

Атрибут HREF задает универсальный локатор ресурса (URL) для внешнего файла описания стилей. Это может быть ссылка на файл с любым именем, а не только на файл с расширением *.css.

8.1.2.4. Импорт описания стилей

Импорт описателей стилей в некотором смысле составляет конкуренцию представленному выше указанию на внешний описатель стиля.

Импортировать стиль можно либо внутрь элемента STYLE, либо внутрь внешнего файла, который представляет собой описатель стиля.

8.1.3. Синтаксис

Формально стиль отображения элементов разметки задается ссылкой в элементе разметки на селектор стиля. Синтаксис описания стилей в общем виде представляется следующим образом:

```
selector[, selector[,... ]]
{ attribute:value;
  [attribute:value;... ] }
```

Или

```
selector selector [selector... ]
{ attribute:value;
  [attribute:value;... ] }
```

8.1.3.1. Селектор - имя элемента разметки

Когда автор Web-узла хочет определить общий стиль всех страниц, он просто прописывает стили для всех элементов HTML-разметки, которые будут использоваться на страницах. Это дает возможность скомпоновать страницы из логических элементов, а стиль отображения элементов описать во внешнем файле.

Такой способ создания сайта позволяет автору изменять внешний вид всех страниц путем внесения изменений в файл описания стилей, а не в файлы HTML-страниц.

8.1.3.2. Селектор - имя класса

Имя класса не является каким-либо стандартным именем элемента HTML-разметки. Оно определяет описание класса элементов разметки, которые будут отображаться одинаково. Для того, чтобы отнести элемент разметки к тому или иному классу, нужно воспользоваться его атрибутом CLASS

8.1.3.3. Селектор - идентификатор объекта

Объектная модель документа (Document Object Model) описывает документ как дерево объектов. Объектами являются: сам документ, его разделы (элемент DIV), картинки, параграфы, приложения и т. п. Каждый из объектов можно поименовать и обращаться к нему по имени. Данная возможность используется при программировании страниц на стороне клиента.

Применение идентификатора объекта оправдано еще и в случае модификации атрибута описания стиля для данного объекта в его CSS-описании. Вместо двух описаний классов, которые отличаются только одним из параметров, можно создать одно описание класса и описание идентификатора объекта. Описание стиля для объекта задается строкой, в которой селектор представляет собой имя этого объекта с лидирующим символом «#».

8.1.4. Наследование и переопределение

При обсуждении технических спецификаций часто бывает полезно вникнуть в смысл названия. В названии принято точно определять суть и назначение стандарта или спецификации. Описание стилей отображения элементов HTML-разметки носит название «Каскадные таблицы стилей». Со словом «стилей» все более-менее понятно. Под словом «таблицы» следует понимать набор свойств элемента разметки, который можно представить в виде строки в таблице свойств, т.е. элементы разметки — строки, а свойства — столбцы. А вот слово «каскадные» требует пояснения.

Во-первых, существует иерархия элементов разметки (дерево объектов на странице). Во-вторых, свойства этих объектов могут наследоваться. Таким образом в дереве объектов образуется ветвь, которая ведет к листу дерева — элементу разметки, например, элементу списка или параграфу. Его свойства

определяются элементами разметки, в которые вложен элемент, и описателями стиля для данного элемента.

Когда объяснение некоторого феномена HTML-разметки растягивается на несколько параграфов, имеет смысл воспользоваться приведенной ниже графической схемой построения страницы.



Рис. 8. 4.

При использовании стилей действуют следующие правила старшинства стилей:

- сначала применяются стили браузера по умолчанию;
- стили браузера по умолчанию переопределяются прилинкованными стилями (элемент LINK заголовка документа);
- прилинкованные стили переопределяются описаниями стилей в элементе STYLE;
- стили элемента STYLE переопределяются атрибутом STYLE в любом из элементов разметки.

8.2. Контрольные вопросы

1. Для чего нужны каскадные таблицы стилей?
2. Перечислите способы применения стилей.
3. Для чего разработаны стили?
4. Для чего нужен элемент STYLE?
5. Как можно импортировать стиль?
6. Какие существуют правила старшинства стилей?
7. Для чего предназначены каскадные таблицы?
8. Перечислите способы применения стилей в CSS.
9. Как осуществляется ссылка на внешнее описание стиля?
10. Все ли атрибуты могут наследоваться? Приведите примеры.
11. Перечислите действующие правила старшинства стилей.

8.3. Задания

1. Используя тег STYLE, измените код имеющейся страницы сайта. Сравните написание кодов HTML и CSS. (R080101)
2. Напишите код, используя селекторы имя класса и идентификатор объекта. (R080102)
3. Используя группирование и наследование, оптимизируйте приведённые стили в селекторах CSS. (R080103)

```
a:link {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  text-decoration: none;
  font-size: 11px;
  color: #3A681A;
}
a:visited {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 11px;
  color: #3A681A;
  text-decoration: none;
}
a:hover {
  text-decoration: none;
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 11px;
  color: #5CA22E;
}
a:active {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 11px;
  color: #DC0000;
}
.pole {
  border: 1px solid #008000;
```

```
width: 95px;
font-size: 11px;
background-color: #E7F2D7;
height: 17px;
}
.pole2 {
border: 1px solid #008000;
font-size: 11px;
background-color: #E7F2D7;
height: 17px;
}
```