

Технология CSS в проектировании Web – сайтов

Методические указания к лабораторной работе

Цель работы: научиться работать с каскадными таблицами стилей при проектировании Web – сайтов.

Теоретическая справка

Дизайн Web-страниц - это точное размещение компонентов HTML - кода относительно друг друга в рабочей области окна браузера.

Недостатки такого определения Web-дизайна очевидны. В нем не учтены ни цвет, ни форма, ни другие свойства компонентов HTML-страниц. Главное в этом определении - показать ограниченность возможностей HTML-разметки. Позиционирование компонентов на странице является одним из самых слабых мест в HTML.

К компонентам страницы относятся:

- *блоки текста,*
- *графика и*
- *встроенные приложения.*

Размер и границы каждого из этих компонентов в рамках HTML-разметки задаются с разной степенью точности. Размер графики и приложений можно задать с точностью до пиксела. Размеры текстовых блоков в HTML задать нельзя: они вычисляются браузером исходя из относительного размера шрифта по умолчанию.

Автор страницы не может заранее определить настройки браузера пользователя, что существенно ограничивает число вариантов представления информации на странице.

Нельзя сказать, что разработчики браузеров не пытались изменить данную ситуацию. В ранних версиях браузеров CERN для платформы NEXT и в браузерах WWWC автор страницы имел возможность переопределять настройки браузера по умолчанию через HTML-разметку. Но этот подход не получил продолжения в коммерческих продуктах.

Другой способ управления настройками браузера - программирование на JavaScript. Бурное развитие этого языка позволяет говорить о возможности полного контроля над процессом отображения HTML-страниц. Недостаток JavaScript - отказ от декларативного характера разметки и относительно большой объем кода для переопределения свойств элементов разметки.

Технология 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.

Таблицы стилей также позволяют определить цвет и начертание текстового фрагмента, изменять эти параметры внутри текстового блока, выполнять выравнивание текстового блока относительно других блоков и компонентов страницы.

Наличие подобных возможностей позволяет говорить о CSS как о средстве разделения логической структуры документа и формы его представления.

Логическая структура документа определяется элементами HTML-разметки, в то время как форма представления каждого из этих элементов задается CSS-описателем элемента.

Термины и определения

Кратко рассмотрим основные термины и определения, касающиеся CSS. Детальные сведения – см. спецификацию <http://loc.stack.ru/projects/CSS2/css2ru.zip>

Таблица стилей

Набор выражений, определяющих представление документа.

Таблицы стилей имеют три различных источника происхождения, относящихся к разработчику, пользователю и агенту пользователя.

Корректная таблица стилей

Корректность таблицы стилей зависит от уровня CSS, используемого для ее создания. Так, корректность таблиц стилей, написанных в CSS1, не изменится при переходе к CSS2. Но из-за внесения некоторых изменений в CSS2 появляются различия в семантике написания ряда таблиц в CSS1 и CSS2.

Исходный документ

Документ, к которому относятся одна или несколько таблиц стилей. С использованием определенной системы кодировки данный документ представляется в виде *дерева элементов*. Каждый элемент представлен своим именем, идентифицирующим его тип; числом его атрибутов (это число не является обязательным) и своим, возможно пустым, содержимым.

Язык документа

Система кодировки, используемая для представления исходного документа (например, HTML или XML).

Элемент

Примитивная синтаксическая конструкция языка документа. Большинство правил, используемых в таблицах стилей CSS, используют имена элементов (например, "P", "TABLE" или "OL" для HTML) для определения информации об их представлении.

Атрибут

Значение, связанное с элементом и состоящее из имени и связанного с ним (текстового) значения.

Содержимое

Содержимое, связанное с элементом в исходном документе. Не у всех элементов имеется содержимое. Элементы могут быть пустыми. В качестве содержимого некоторого элемента может выступать текст или множество вложенных элементов, относительно которых исходный элемент называется родительским.

Генерируемое содержимое

Содержимое элемента, которое было связано с ним в результате использования соответствующих таблиц стилей. Генерируемое содержимое заменяемого элемента поступает не из исходного документа. Это может быть дополнительный текст для элемента (например, значение атрибута "alt" в HTML) и элементы, явно или неявно привносимые таблицей стилей, например, маркеры, номера и др.

Дерево документа

Дерево элементов, закодированное в исходном документе. Каждый элемент в этом дереве имеет только один родительский элемент, за исключением корневого элемента, у которого его нет. Ниже приводится пример исходного документа на языке HTML:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
  <TITLE>Моя домашняя страница</TITLE>
  <BODY>
    <H1> Моя домашняя страница;/H1>
    <P>Добро пожаловать на мою домашнюю страницу! Позвольте рассказать вам
о моих любимых композиторах:
    <UL>
      <LI>Элвис Костелло
      <LI>Иоган Брамс
      <LI>Джорж Брассенс
    </UL>
  </BODY>
</HTML>
```

В результате получается следующее дерево:

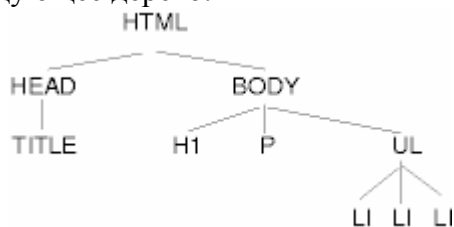


Рис.1 Дерево HTML - документа

Дочерний элемент

Элемент А называется дочерним по отношению к элементу В, если и только если В является родительским элементом элемента А.

Потомок

Элемент А называется потомком элемента В, если выполняется одно из следующих условий:

- (1) А является дочерним элементом элемента В;
- (2) А является дочерним элементом элемента С, который, в свою очередь, является потомком В.

Предок

Элемент А называется предком элемента В, если и только если В является потомком элемента А.

Сестринский элемент

Элемент А называется сестринским элементом элемента В, если и только если элементы А и В являются дочерними элементами одного и того же элемента. Элемент А является предшествующим сестринским элементом, если в дереве документа он находится перед элементом В. Элемент А является следующим сестринским элементом, если в дереве документа он находится после элемента В.

Разработчик

Разработчик - это человек, создающий документы и связанные с ними таблицы стилей. Документы и ассоциированные с ними таблицы стилей генерируются средством разработки.

Пользователь

Пользователь - это человек, взаимодействующий с агентом пользователя с целью просмотра, прослушивания или другого использования документа и связанных с ним

таблиц стилей. Пользователь может создать персональную таблицу стилей, в которой будут закодированы персональные настройки.

Агент пользователя

Агент пользователя - это произвольная программа, которая интерпретирует документ, написанный на некотором языке структуризации, и применяет связанные с ним таблицы стилей в соответствии с правилами данной спецификации. Агент пользователя может выводить документ на экран, осуществлять его звуковое воспроизведение, выводить на печать, конвертировать в другой формат и т.д.

Способы применения CSS

Под способами применением CSS понимаем форму декларирования стиля на HTML-странице и форму связывания описания стиля отображения элемента разметки с самим этим 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.

Различают четыре способа применения стилей:

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

Переопределение стиля

Переопределение стиля в элементе разметки - это применение атрибута STYLE у данного элемента разметки:

```
<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 отчеркивание):

```
<hr style="width:100px;">
```

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

Элемент STYLE

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

Элемент STYLE позволяет определить стиль отображения для:

- стандартных элементов HTML-разметки
- произвольных классов (селектор class)

- HTML-объектов (селектор id)

Стандартные элементы разметки описываются в элементе STYLE следующим способом:

```
<head>
<style>
p {color:darkred;text-align:justify;font-size:8pt;} </style>
</head>
<body>
```

...

```
<p>
```

Этот параграф используется в качестве примера применения описания стиля для стандартного элемента HTML-разметки.

```
</p>
```

...

```
</body>
```

Теперь все параграфы документа будут отображаться стилем из элемента STYLE, если только стиль не будет переопределен каким-либо способом. В STYLE можно определить стиль любого элемента разметки.

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 из себя файл, содержание которого - 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://kuku.ru/my_css.css">
```

Пример от студии “Квадрат”:

```
<LINK HREF="style.css" REL="stylesheet" TYPE="text/css">
```

Это в теге Head:

```
<HEAD>
<TITLE>О Студии. Дизайнерская студия "Квадрат"</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=windows-1251">
<LINK HREF="style.css" REL="stylesheet" TYPE="text/css">
<META NAME="keywords" CONTENT="дизайнерская студия квадрат, дизайнерская студия, дизайн, веб-дизайн, сайт, разработка сайтов, сайты на заказ, креативф :)) ">
<META NAME="description" CONTENT="Дизайнерская студия <Квадрат>. Создание и разработка сайтов.">
<SCRIPT language=JavaScript1.2 src="main.js"></SCRIPT>
</HEAD>
```

Важными здесь являются значения атрибутов rel и type. Rel обязан иметь значение "stylesheet". Type может принимать значения: "text/css" или "text/javascript". Второй тип описания стилей введен Netscape.

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.

Импорт описания стилей

Импорт описателей стилей - это в некотором смысле конкурент описанному выше указанию на внешний описатель стиля. Импортировать стиль можно либо внутрь элемента STYLE, либо внутрь внешнего файла, который представляет собой описатель стиля. Оператор импорта стиля должен предшествовать всем прочим описателям стилей:

```
<style>  
@import:url(http://kuku.ru/style.css)  
a {color:cyan;text-decoration:underline;}  
</style>
```

Импортируемый стиль можно переопределить либо через описатель элемента в STYLE, либо через атрибут элемента style.

Редактор стилей системы DreamWeaver

Система DreamWeaver – третья, завершающее звено в технологии Web – дизайна, имеет развитые средства поддержки стилей. В основном, они реализованы в редакторе стилей (CSS Styles).

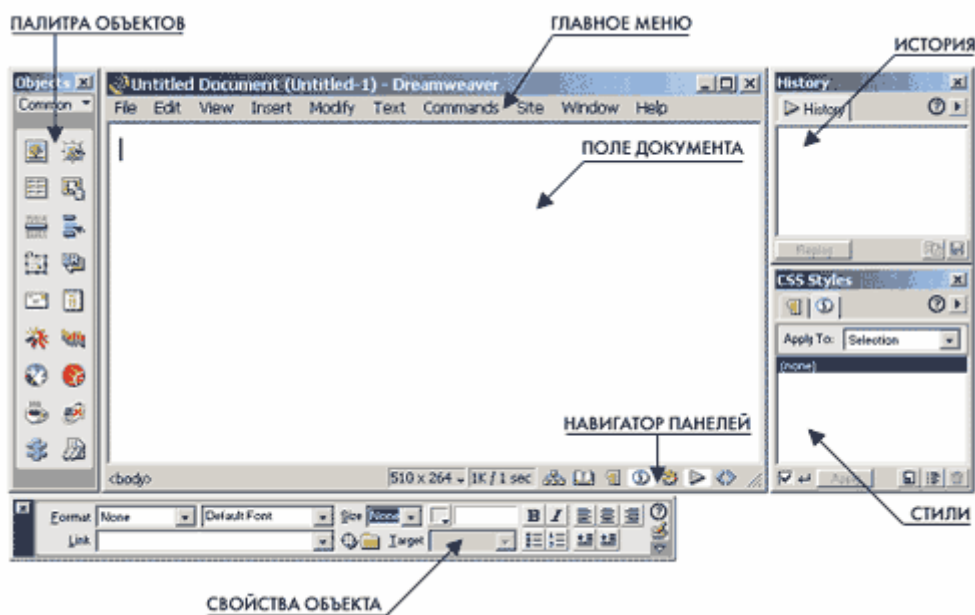


Рис. 1. Рабочее окно программы

Запуск редактора стилей. CSS Styles (CSS – стили) находятся в правой нижней части окна DreamWeaver, как показано на рис. 2.

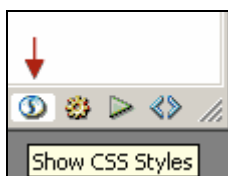


Рис. 2. Запуск редактора стилей из окна HTML-документа.

Возможности редактора стилей. В главном окне редактора стилей помещены ссылки на имеющиеся файлы стилей. Для создания нового стиля нажимаем New Style, как показано на рис. 3.

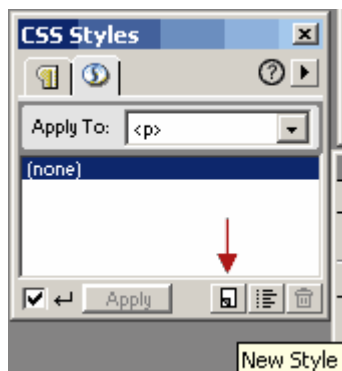


Рис. 3. Создание нового стиля.

В диалоге выбираем тип Make Custom Style (class) и вводим название стиля.

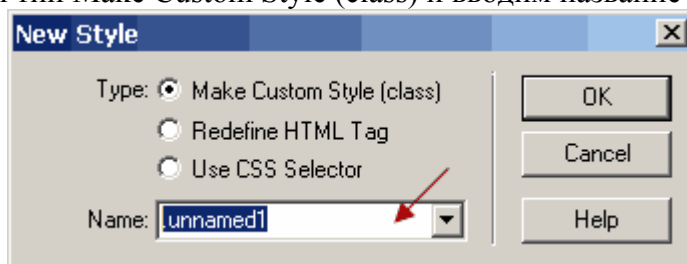


Рис. 4. Окно создания нового стиля.

На рис. 5 показано окно редактирования стилей.

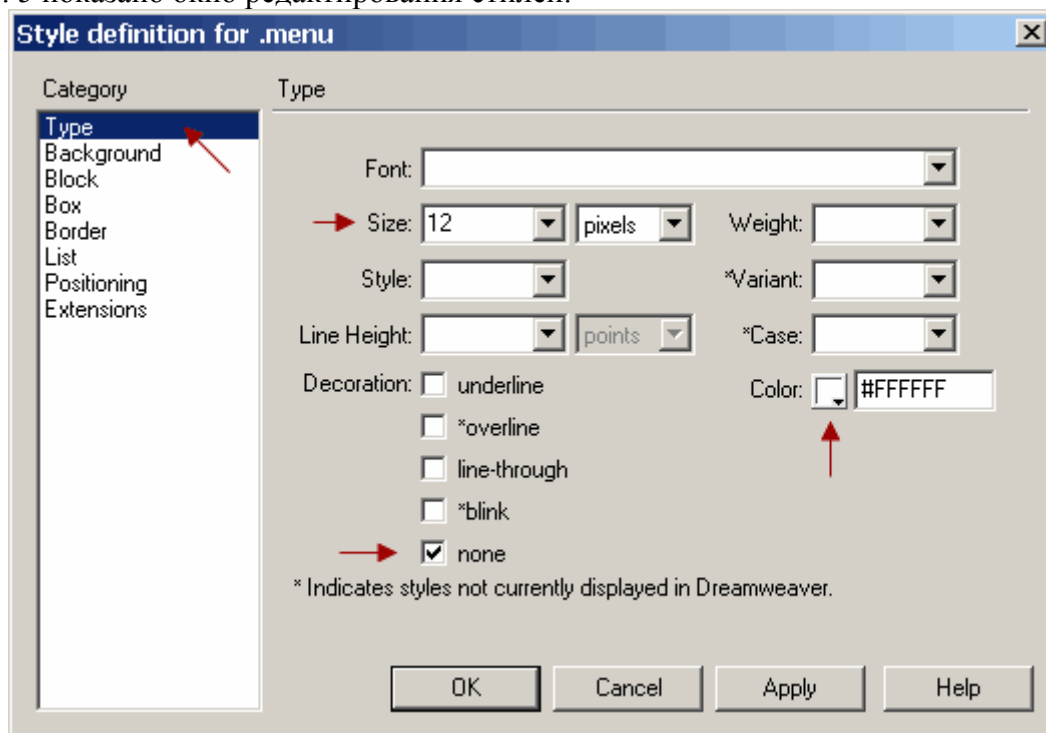


Рис.5. Окно редактора стилей.

В результате в CSS Styles появится созданный вами стиль. На рис. 6 показан пример применения стиля на параграф текста. Для этого выделяем нужный фрагмент, нажимаем

на букву <P> в нижнем левом углу окна и в панели стилей просто выбираем наш стиль (menu).

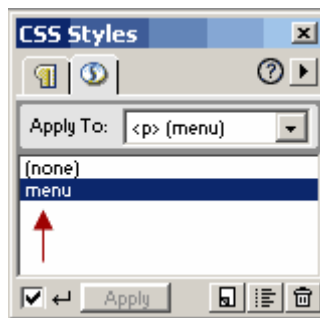


Рис.6. Окно CSS Styles.

Новый стиль можно сразу назначить тегам

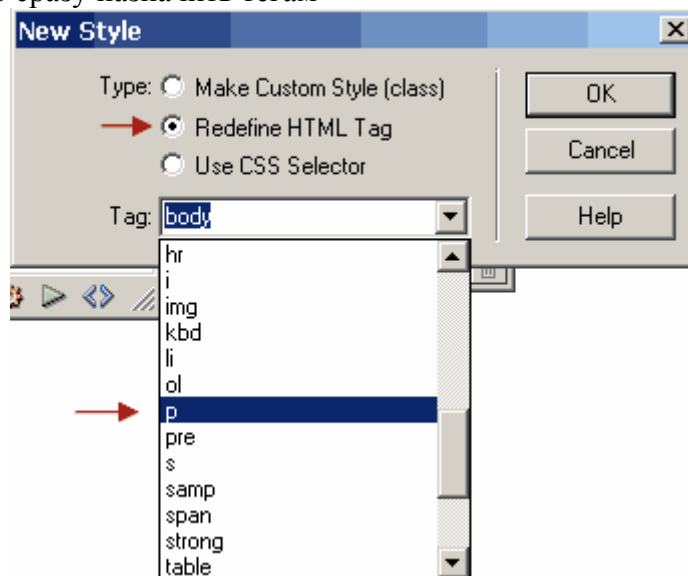


Рис.8. Окно сознания нового стиля тегов.

или ссылкам.

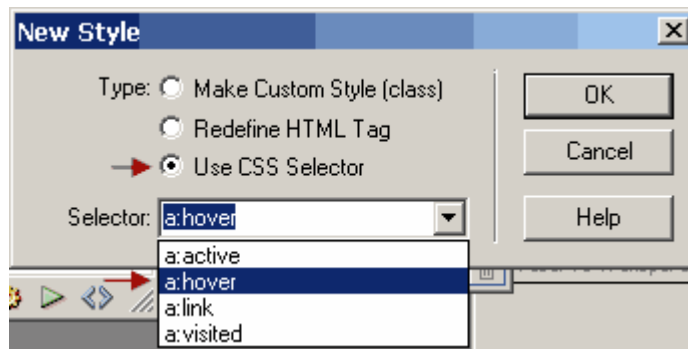


Рис.8. Окно создания нового стиля ссылок.

Таким образом, редактор стилей является мощным и гибким инструментом создания и редактирования стилей в технологии CSS.

Программное обеспечение и оборудование для выполнения работы

Работа выполняется на современном персональном компьютере. Необходимое программное обеспечение:

- *Навигатор Web-страниц (браузер)*
- *Система DreamWeaver*
- *Web-сайт в виде набора HTML-страниц и файлов каскадных таблиц стилей.*

Порядок выполнения работы

1. Ознакомиться с теоретической справкой, спецификациями HTML и CSS2.
2. Ознакомиться с примером Web-сайта. Определить единые дизайнерские решения для всех страниц сайта: шрифты, расположение текста, цветовые решения.
3. Запустить систему DreamWeaver.
4. Открыть в системе DreamWeaver страницу сайта. Начать желательно с главной страницы.
5. В редакторе CSS системы DreamWeaver открыть файл каскадных таблиц стилей данного сайта.
6. Ознакомиться с описаниями стилей, содержащимися в файле.
7. Найти в коде HTML-разметки открытой страницы ссылки на описания стилей.
8. Пункты 4 – 7 выполнить для других страниц сайта. Сделать выводы о способах применения описания стилей, использованных в исследуемом сайте.
9. По согласованию с преподавателем изменить один или несколько стилей. Оценить полученные результаты.
10. Разработать и включить в сайт новый стиль. Оценить полученные результаты.
11. Ответить на контрольные вопросы.

Контрольные вопросы

1. Почему таблицы стилей называются каскадными?
2. К какому типу языков относится CSS - это алгоритмический язык, язык скриптов или язык разметки?
3. Какие существуют способы применения стилей?
4. Как раскрывается аббревиатура CSS?
5. Каким образом единственный файл стилей определяет представление текстов в нескольких Web – страницах?
6. Как, не имея системы DreamWeaver, составить и применить файл стилей?
7. Допустимо ли применение нескольких файлов стилей для одного сайта?
8. Изменяет ли применение CSS общий стиль проектирования сайтов по сравнению с технологией DreamWeaver?
9. В чем преимущества и недостатки технологии CSS?