**Структура HTML документа**

**HTML элементы** являются основными составляющими веб страницы, с помощью них осуществляется все форматирование. В данном уроке рассмотрена **структура HTML элемента**, которая является одинаковой почти для всех элементов веб страницы, которые могут иметь атрибуты. Структура **HTML элемента** включает имя атрибута, значение атрибута, содержимое тега.

Как мы уже говорили в предыдущих темах оформление web-страниц можно осуществлять двумя способами:

* первый – средствами ***таблиц стилей CSS*** (более прогрессивный метод),
* второй – средствами ***атрибутов у тегов HTML***.

Рассмотрим второй способ.

**1. Структура HTML-элемента**

HTML-элемент также имеет свой синтаксис.

**<[имя тега] [имя атрибута 1]=["значение атрибута 1"] [имя атрибута 2]=["значение атрибута 2"] ...   [имя атрибута N]=["значение атрибута N"]>Содержимое под действием тега</[имя тега]>**

**Кавычки вокруг значения атрибута обязательны в тех случаях, когда значение это содержит какие-либо символы, кроме букв, цифр, точки или дефиса, но с точки зрения совместимости с XML, лучше пользоваться кавычками всегда.**

Многие теги имеют атрибуты со значениями, изменяющие действие тега.

*Например:*

**<h1 align="center">Пример заголовка</h1>**

Разберем этот тег с одним атрибутом согласно синтаксису (рисунок 1)

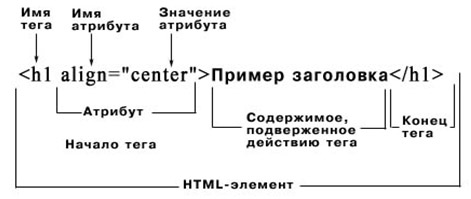


Рисунок 1. Синтаксис HTML-элемента

Текст, помещенный внутри тегов **<h1 align="center">Текст</h1>,** будет выглядеть, как центрированный заголовок.

На рисунке 2 приведен вид этой HTML-страницы в браузере.

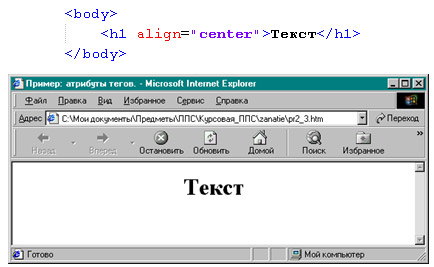


Рисунок 2. Пример использования атрибута align

Атрибут align указывает выравнивание элемента, в данном случае заголовка. Возможные значения данного атрибута представлены в таблице 1.

**Таблица 1**

|  |  |
| --- | --- |
| align="right" | Выравнивает элемент по правому краю |
| align="left" | Выравнивает элемент по левому краю |
| align="center" | Выравнивает элемент по центру |
| align="justify" | Выравнивает элемент по ширине |

В большинстве случаев атрибуты являются необязательными, но в их отсутствие интерпретатор HTML должен использовать значения по умолчанию (так называемый ***стиль браузера***).

Все атрибуты тегов есть в справочнике **Sprav\_ HTML.doc** в папке **CD/Справочник HTML**.

**Регистр букв в идентификаторах тегов и атрибутов и значениях атрибутов не учитывается.**

**2. Тело документа**

Между тегами <body> и </body> размещается основная часть материала документа.

**Атрибуты тега <body>**

Таблица 2

|  |  |
| --- | --- |
| background | Атрибут указывает URL-адрес изображения, которое следует использовать в качестве фона документа |
| bgcolor | Определяет цвет фона документа |
| bgproperties | Если этот атрибут имеет значение fixed, то изображение фона не будет прокручиваться |
| leftmargin | Атрибут определяет ширину левого поля в пикселях |
| topmargin | Атрибут определяет ширину верхнего поля в пикселях |
| text | Атрибут определяет цвет текста |

Все атрибуты можно просмотреть в файле **Sprav\_ HTML.doc** в папке **CD/Справочник HTML**.

 Например:

         <body background="fons/image.jpg">

В данном примере фоном документа будет являться рисунок **image.jpg**, который находится в папке **fons**. При этом HTML-документ находится в файловой структуре на том же уровне, что и папка **fons**.

**3. Элементы h1, h2, h3, h4, h5, h6**

Структурирование тела документа выполняется элементами h1, h2, h3, h4, h5, h6.

Элементы заголовков являются парными, поэтому должны имеет открывающий <h1> и закрывающий </h1> теги.

HTML располагает шестью уровнями заголовков: h1 (самый верхний), h2, h3, h4, h5 и н6 (самый нижний). Функции элементов заголовков аналогичны заголовкам в текстовых редакторах.

**Практическое задание 1**

1. Откройте файл **shablon.html** в Notepad++ и cохраните под новым именем **elementH.html**в папке **myhouse**.
2. Измените текст, как в листинге на рисунке 3. В качестве фона используйте изображение **fon10.jpg**из папки **html\_css\_3.**
3. Запустите файл **elementH.html**

На рис. 4 показано, как разные уровни заголовков выводятся браузером.

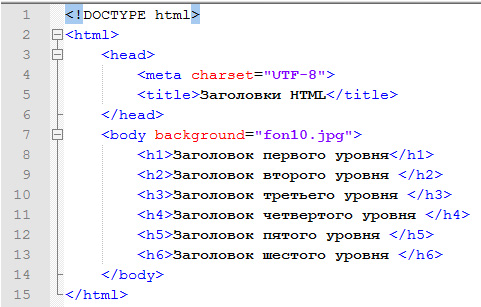


Рисунок 3

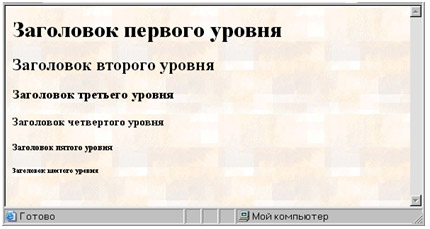


Рисунок 4. Разные уровни заголовков

**Практическое задание 2**

1. Откройте файл **elementH.html** в Notepad++ и cохраните документ в папку под новым именем **align.html**
2. Измените код как в листинге на рисунке 5 (для фона можете использовать любой цвет из папки **colors).**
3. Запустите файл **align.html**

Ваш документ должен выглядеть примерно так, как показано на рис. 6.

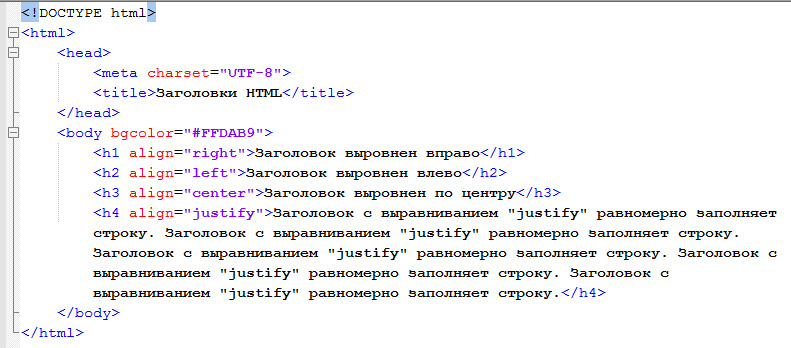


Рисунок 5

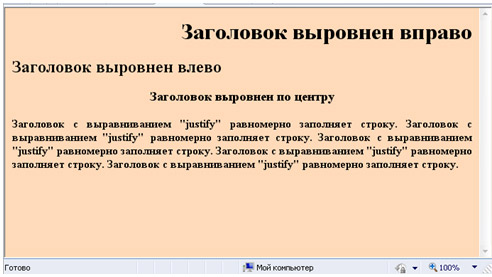


Рисунок 6. Выравнивание заголовков

**4. Работа с текстом**

***Разделение текста на абзацы***

Тег <p> задает начало абзаца и вставляет отступы между абзацами. Тег абзаца имеет только один атрибут align.

align="left" – выравнивание абзаца по левому краю текста (значение по умолчанию).

align="right" – выравнивание абзаца по правому краю текста.

align="center" – центрирование абзаца.

align="justify" – выравнивание абзаца по ширине.

***Принудительный разрыв строки***

Повторим, что тег <br /> позволяет выполнить перенос оставшейся части текста абзаца на следующую строку. Это непарный тег и в отличие от тега абзаца не увеличивает интервал между строками.

***Изменение шрифта***

Тег <font> и </font> позволяет изменить для фрагмента текста тип, размер и цвет шрифта.

**Атрибуты тега <font>**

Таблица 3

|  |  |
| --- | --- |
| size | Атрибут size="n" используется для задания абсолютного размера между 1 и 7. size="+n" или –n – увеличивает или уменьшает шрифт относительно текущего значения. |
| face | Атрибут face="ИМЯ" – задает гарнитуру шрифта, используемого для отображения данного фрагмента текста. Можно через запятую задать несколько имен шрифтов, которые браузер ищет в локальной системе. Использоваться будет тот, чье имя найдено первым.  Например: *<font face="Comic Sans MS,Tahoma,Arial">* |
| color | Атрибут color="Значение" задает цвет текста. |

***Начертания шрифтов***

Тег <b>…</b> – позволяет отобразить текст полужирным шрифтом.

Тег <i>…</i> – позволяет отобразить текст в курсивном начертании.

Тег <u>…</u> – отображает подчеркнутый текст.

**Практическое задание 3**

1. Откройте файл **align.html**и сохраните под новым именем **format\_text.html**в папке **myhouse.**
2. Измените код как в листинге на рисунке 7 (для фона можете использовать любой цвет из папки **colors).**Чтобы не набирать текст вручную, Вы можете cкопировать этот текст из файла **format\_text.txt** из папки**html\_css\_3**
3. Запустите файл **format\_text.html**

Ваш документ должен выглядеть примерно так, как показано на рис. 8.

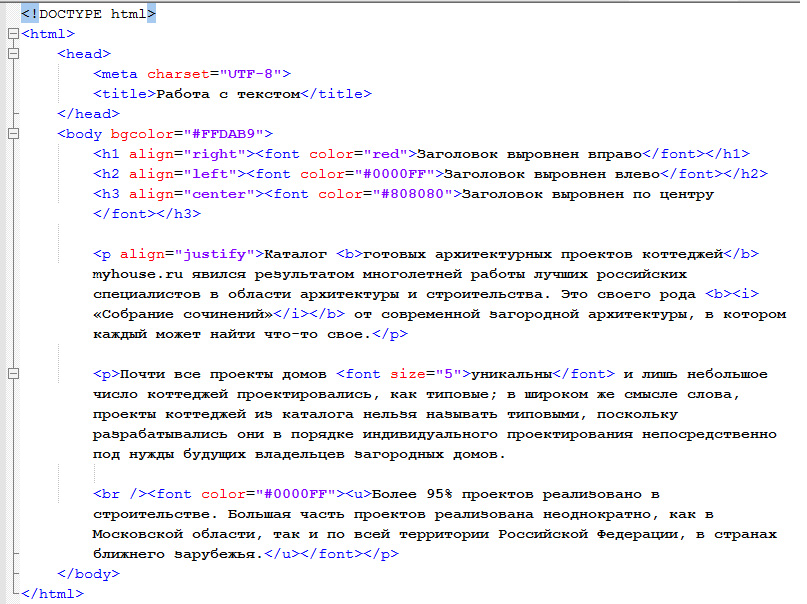


Рисунок 7

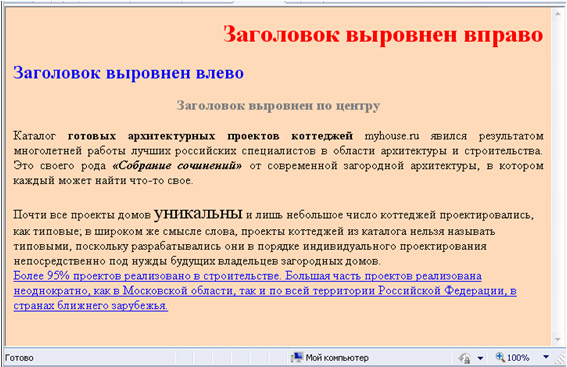


Рисунок 8. Форматирование текста

**5. Горизонтальные линии на HTML-странице**

С помощью тегов <P> и <BR /> выделяются смысловые фрагменты в документе. Но иногда необходимо отчетливо показать границу между разделами, чтобы подчеркнуть жесткое разделение текста, это можно сделать с помощью горизонтальных линий.

Горизонтальная линия организуется с помощью тега <HR>.

Атрибуты тега <HR> определяют выравнивание линии по горизонтали, позволяют изобразить линии трехмерными или подавить иллюзию трехмерности, задать толщину и длину линии. Горизонтальные линии применяются для:

1. Разделения текстовых фрагментов. С помощью тонких линий можно визуально разделить части текста.
2. Выделения заголовков, сносок, ремарок автора;
3. Создания прямоугольников и квадратов. Путем подбора атрибутов SIZE и WIDTH горизонтальную линию можно превратить в прямоугольник или квадрат.

 Пример:

<hr size="4" width="75%" align="center" />

Браузер изобразит горизонтальную линию толщиной 4 пикселя и шириной, составляющей 75% от ширины экрана браузера (рис.9).

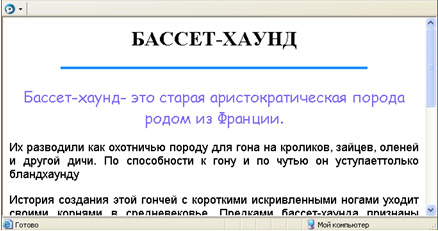


Рисунок 9. Пример горизонтальной линии на web-странице

**Практическое задание 4**

1. Откройте раздел «Горизонтальные линии» в файле справочника «Sprav\_html.doc». Ознакомьтесь с атрибутами тега <HR>.
2. В файле **main.html** создайте разделительные полосы между смысловыми группами и вверху страницы вставьте изображение по центру (рисунок 10).
3. Сохраните файл.



Рисунок 10

**6. Бегущая строка**

Эффект «бегущей строки» – популярное украшение web-страниц.

Текст, выделенный тегами <marquee> и </marquee>, изображается браузером в виде бегущей строки.

**Практическое задание 5**

1. Откройте файл справочника «Sprav\_html.doc». Найдите раздел «Бегущая строка». Ознакомьтесь с атрибутами всех тегов, используемых при создании бегущей строки. Будьте готовы ответить на вопросы по правилам задания значений этих атрибутов.
2. Разберите приведенный в листинге на рисунке 11 программный код. Реализуйте его.
3. Сохраните под именем **stroka.html**

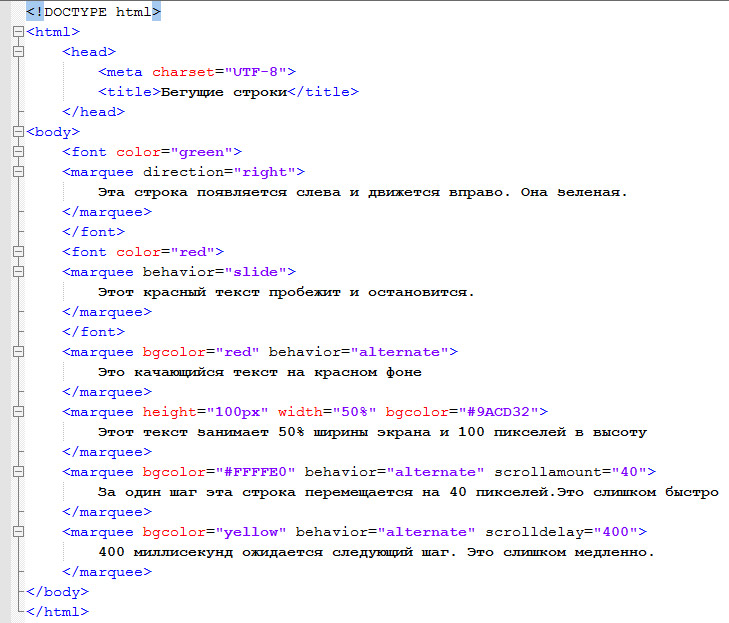
******

Рисунок 11. Бегущие строки

**Практическое задание 6**

1. Создайте бегущую строку для будущего сайта myhouse.ru примерно так, как изображено на рис. 12.
2. Сохраните в папке **public\_html** под именем **shapka.html**

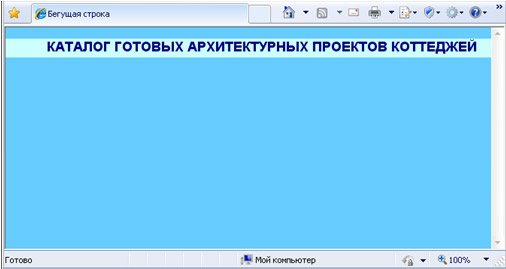


Рисунок 12. Бегущая строка для сайта MyHouse.ru

**7. Формирование таблиц**

С помощью таблиц можно различным образом компоновать текст и графику на странице.

1. Основные теги **<table>** и **</table>** – начало и конец таблицы.
2. Теги **<tr>** и **</tr>**. Эти теги окружают каждую строку таблицы. Два набора тегов <tr> – две строки, восемь наборов – восемь строк.
3. Теги **<td>** и **</td>**. Эти теги определяют отдельную ячейку строки. В них содержатся данные таблицы. Число ячеек в строках таблицы не обязательно должно быть постоянным. В одной строке может быть пять ячеек, в другой три.
4. Теги **<th>** и **</th>**. Этот тег используется для задания заголовков столбцов и строк таблицы. Текст, заключенный между этими тегами, автоматически записывается жирным шрифтом и располагается посередине ячейки.
5. Теги **<caption>** и **</caption>** – создают заголовки таблицы. По умолчанию заголовки центрируются и размещаются либо над **(<caption align= "top">),** либо под **(<caption align= "bottom">)** таблицей. Заголовок может состоять из любого текста и изображения.

Пример HTML-кода таблицы (рисунок 13) и вид этой таблицы в браузере приведены на рис. 1

**Практическое задание 7**

1. Реализуйте код, приведенный в листинге на рисунке 13.
2. Сохраните файл под именем **table.html**

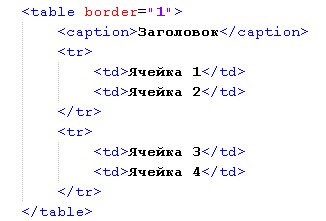
******

Рисунок 13. Простая таблица

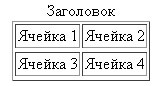


Рисунок 14. Вид простой таблицы в браузере

**Объединение ячеек**

Для объединения столбцов таблицы в теге ячейки **td** применяется атрибут **colspan**.

Для объединения строк таблицы в теге ячейки **td** применяется атрибут **rowspan**.

Пример объединенных столбцов приведен на рис. 15, объединенных строк на рис. 16.

**Практическое задание 8**

1. В файле **table.html** реализуйте коды из листингов на рисунках 15 и 16.
2. Сохраните файл.

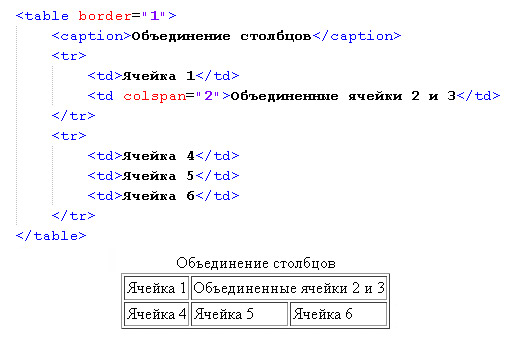


Рис. 15. Вид простой таблицы с объединенными столбцами

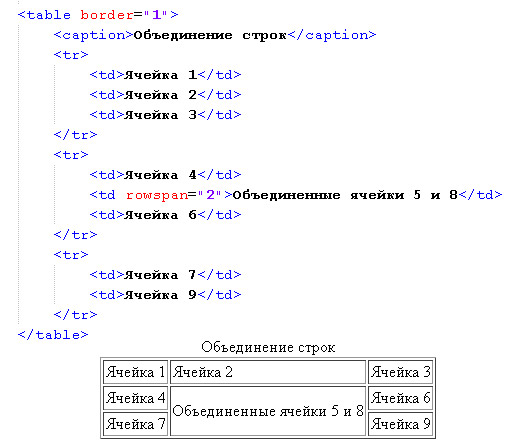


Рис. 16. Вид простой таблицы с объединенными строками

Действие всех перечисленных тегов можно уточнить с помощью атрибутов табличных тегов.

**Практическое задание 9**

Откройте файл справочника **Sprav\_html.doc** в папке **CD**. Найдите раздел «Форматирование таблиц в HTML». Ознакомьтесь с атрибутами всех тегов, необходимых для создания таблицы.

**Большие таблицы**

1. Создайте web-страницу, содержащую таблицу с расписанием занятий на четную и нечетную недели. Структура, содержание и вид таблицы в браузере приведены на рис. 17.

2. При создании таблицы:

* Для заголовка нечетной недели задать красный фон, для четной – синий;
* Рамку таблицы сделать цветной;
* В заголовок таблицы включить рисунок из папки **CD/html\_css\_3**;
* Растянуть последнюю строку таблицы на все ячейки соответствующей недели (рис. 17).



Рис. 17

3. Измените код программы так, чтобы ячейка, содержащая текст «В расписании возможны изменения» была общей для обеих недель.

4. Оформите содержимое последней строки таблицы «В расписании возможны изменения» бегущей строкой. Параметры бегущей строки: направление движения бегущей строки справа налево, цвет фона бегущей строки – светло-желтый, повторение бегущей строки – 5 раз.

5. Сохраните файл под именем **raspisanie.html**(рис. 18).

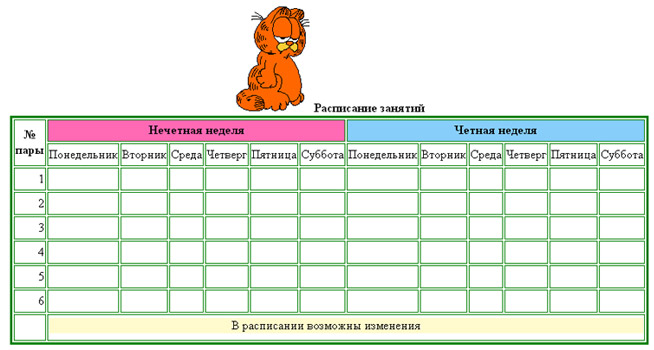


Рисунок 18. Расписание занятий

**Практическое задание 10**

Для нашего сайта необходимо оформить web-страницу с информацией о проектах таунхаусов в виде таблицы. Всего на этой странице должно быть 9 проектов. Материалы (фотографии и тексты) можно найти в папке **CD/** **html\_css\_3/townhouse**.

1. Создайте в папке **public\_html** папку **kategoria**.
2. Внутри папки **kategoria** создайте еще папку **townhouse.**
3. Внутри папки **townhouse** создайте еще папку **img\_townhouse.**
4. Скопируйте в папку **img\_townhouse**все необходимые изображения. Они находятся в папке **CD/townhouse.**
5. Сохраните страницу под именем **townhouse.html**в папку **townhouse.**

**Внутренние стили**

Последним вариантом использования стилевых таблиц является применение их к конкретному HTML-элементу при помощи атрибута STYLE, который разрешен практически для каждого элемента.

Например*: <h1 style="color:blue; font-family:arial;">*.

**Практическое задание 11**

1. На web-странице townhouse.html для текста под заголовком примените к тегу абзаца внутренний стиль, который опишите при помощи *атрибута* STYLE: имя шрифта – Verdana, цвет шрифта – 0000FF, размер шрифта – 10pt, начертание – полужирное.
2. Проверьте работоспособность внутреннего стиля.

 Примерное оформление информации приведено на рис. 19.



Рис. 19. Примерное оформление информации о проектах