



# Язык HTML

# Проектирование и разработка Web-сайтов

- Веб-сайт — это совокупность логически связанной гипертекстовой информации, оформленной в виде отдельных страниц и доступной в сети Интернет, другими словами, Веб-сайт — это совокупность программных, информационных, а также медийных средств, логически связанных между собой.
- Web-страницы или гипертекстовые документы представляют собой текст, в котором содержатся специальные команды, называемые тегами (**tags**). Эти теги обеспечивают форматирование элементов страницы и позволяют размещать на ней графические объекты, рисунки, гиперссылки и т.д.
- Для подготовки Веб-страниц к публикации в Интернете применяется специальный "язык гипертекстовой разметки" —  
**HTML (Hyper Text Markup Language).**
- Разметка гипертекста указывает браузеру, в каком виде информация должна выводиться на экран.
- *HTML не является языком программирования, он только управляет расположением информации в окне браузера.*

# ***HTML***

***HTML*** позволяет формировать на странице сайта текстовые блоки, включать в них изображения, организовывать таблицы, управлять отображением цвета документа и текста, добавлять в дизайн сайта звуковое сопровождение, организовывать гиперссылки с контекстным переходом в другие разделы сервера или обращаться к иным ресурсам Сети и компоновать все эти элементы между собой. Файлы, содержащие гипертекстовый код, имеют расширение `.htm` или `.html`.

# HTML

*HTML-документом* называется файл, содержащий код **HTML**.

Одной из основных функциональных особенностей, характерных именно для данного языка, благодаря которой он и получил свое название, является гиперссылка.

*Гиперссылка* (Hyperlink) – базовый функциональный элемент html-документа, представляющий собой реализацию динамической связи какого-либо объекта данной web-страницы с контекстным содержимым другого документа.

**HTML** – язык не транслируемый, но интерпретируемый. Это означает, что для исполнения готового кода его не нужно компилировать, встроенный в специальное программное обеспечение, предназначенное для просмотра web-страниц, интерпретатор «компилирует» код непосредственно в процессе открытия документа. При этом, обнаружив в тексте ошибку, такая программа не выдает соответствующего предупреждения (если страница не содержит встроенных скриптов Java), а попросту игнорирует всю «ошибочную» строку.

# Браузер

*Браузер* – специальная клиентская программа, предназначенная для просмотра содержимого web-узлов и отображения документов HTML. Браузеры содержат встроенный транслятор языка разметки гипертекста, компилирующий html-код в процессе открытия web-страницы.

Существует несколько классов браузеров, различающихся в основном спектром реализуемых ими возможностей. Два основных класса – это браузеры, поддерживающие и не поддерживающие отображение графических элементов web-страниц. Большинство современных браузеров относятся к первому классу. Наиболее распространены среди пользователей Интернета браузеры Mozilla Firefox, Opera, Internet Explorer, Chrome.

- На данный момент наиболее популярным и продвинутым текстовым редактором является **Notepad++**, который можно найти по адресу <http://notepad-plus-plus.org/>.

К его преимуществам можно отнести бесплатность, подсветка тегов html. В дальнейшем будем ориентироваться именно на этот текстовый редактор.

- Также стоит упомянуть кроссплатформенный текстовый редактор [Visual Studio Code](#). Данный редактор обладает несколько большими возможностями, чем Notepad++, и кроме того, может работать не только в ОС Windows, но и в MacOS и в операционных системах на основе Linux.

Macromedia Dreamweaver 8 - [Untitled-1 (XHTML)]

Файл Редактировать Вид Вставить Изменить Текст Команды Сайт Окно Помощь

▼ Insert Общее Макет Формы Текст HTML Приложение Элементы Flash Избранное



Untitled-1

Код Раздельно Дизайн

Заголовок: Документ Без Имени



```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
5 <title>Документ Без Имени</title>
6 </head>
7
```

▼ CSS

CSS Styles Layers

Все Текущий

Общие для Выбранного

CSS настройки нельзя применить для текущего выбора. Пожалуйста, выберите элементы стиля, чтобы применить CSS настройки.

Правила

Настройки

C:\Documents and Settings\Admin\Рабочий стол\Абзац.html - Notepad++

Файл Правка Поиск Вид Кодировки Синтаксис Опции Макросы Запуск Плагины Окна ?



change.log Абзац.html цвет\_фона.html форматирование.html специальные символы.html

```
1 <!-- форматирование кода с отступами -->
2 <html>
3   <head>
4     <title> Мой первый документ </title>
5   </head>
6   <body>
7     <h1> Мой первый HTML-документ </h1>
8     <hr>
9     <h2>Евгений Онегин</h2>
10    <p>А.С. Пушкин (отрывок)</p>
```

AkeIPad

Файл Правка Найти Кодировки Вид Избранное Настройки Плагины

Окно Справка



```
1 <html>
2 <head>
3   <title> Мой первый документ </title>
4 </head>
5 <body>
6   <h1> Мой первый HTML-документ </h1>
7   <hr>
8   <h2>Евгений Онегин</h2>
9   <p>А.С. Пушкин (отрывок)</p>
10  <p>Мой дядя самых честных правил,
11  <br>Когда не в шутку занемог,
12  <br>Он прилежно, себя самовос...
```



# Структура HTML-документа



# Теги

*Директивы HTML называются тегами. Теги заключаются в угловые скобки. Большинство тегов – парные. Закрывающий тег отличается от открывающего наличием символа /. Синтаксис записи тега:*

*<ТЕГ> обрабатываемое значение </ТЕГ>*

*Теги допускают вложение друг в друга. Все объекты не заключенные в угловые скобки, воспринимаются как текстовые элементы.*

*HTML не чувствителен к регистру, однако рекомендуется применять при написании тегов заглавный регистр для улучшения читаемости кода.*

*Тег – это команда html, указывающая интерпретатору браузера, каким образом он должен обрабатывать соответствующее каждой конкретной директиве значение. Это значение называется **атрибутом тега**. Тег может иметь атрибут или не иметь его. Тег верхнего уровня <HTML> не имеет атрибутов.*

# Теги

Синтаксис записи тега в совокупности с его атрибутом:

**<ТЕГ имя\_атрибута\_1="значение" ... имя\_атрибута\_n="значение">**

Значения атрибутов заключаются в прямые кавычки `""`. Если внутри атрибута тега встречается другое значение, заключенное в кавычки, то есть имеет место вложение одних кавычек в другие, в качестве внутренних кавычек рекомендуется использовать одинарные. Такое выражение выглядит следующим образом:

**<ТЕГ имя\_атрибута\_1="значение\_1; 'значение2'; значение3">**

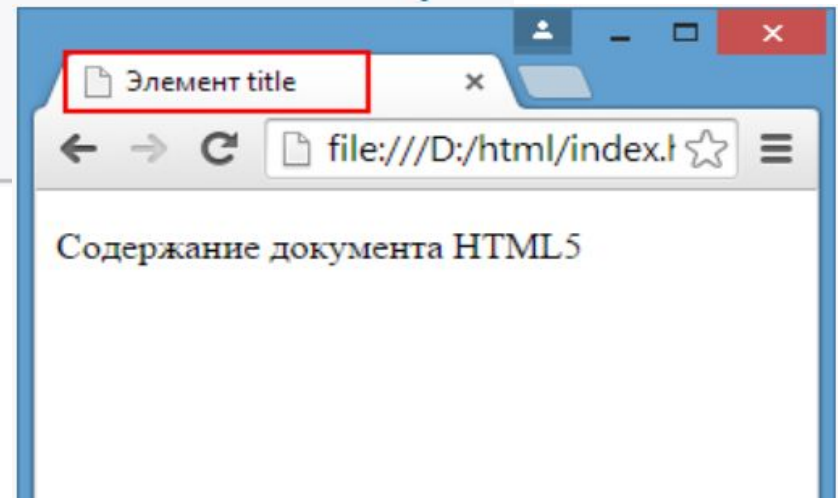
Спецификация языка позволяет опускать кавычки в следующих видах атрибутов:

- ❖ *атрибуты, записываемые только строчными или заглавными символами латинского алфавита и не включающие иных символов, например цифр.*
- ❖ *атрибуты, состоящие только из цифр от 0 до 9.*
- ❖ *атрибуты, обозначающие промежутки времени.*

Например, запись атрибутов `border="1"` или `align="center"` допустимо представить в виде `border=1` или `align=center`.

# Структура HTML документа

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Элемент title</title>
  </head>
  <body>
    <p>Содержание документа HTML5</p>
  </body>
</html>
```



- Для создания документа HTML5 нам нужны в первую очередь два элемента: **DOCTYPE** и **html**. Элемент **doctype** или Document Type Declaration сообщает веб-браузеру тип документа.

**<!DOCTYPE html>** указывает, что данный документ является документом html и что используется html5, а не html4 или какая-то другая версия языка разметки.

А элемент **html** между своим открывающим и закрывающим тегами содержит все содержимое документа.

- Основная часть документа html, фактически все, что увидим в своем браузере при загрузке веб-страницы, располагается между тегами **<body>** и **</body>**. Здесь размещаются большинство элементов html.

- Хотя большинство элементов в HTML5 остаются теми же, что и в ранних версиях, но несколько изменился способ их использования. Рассмотрим базовые элементы HTML, их предназначение и использование.

- **Элемент head и метаданные веб-страницы**

Одним из первых элементов html-документа является элемент **head**, задача которого состоит в установке метаданных страницы и ряда сопроводительной информации. Метаданные содержат информацию о html-документе.

- **Заголовок**

Для установки заголовка документа, который отображается на вкладке браузера, используется элемент **title**

# СТРУКТУРА HTML ДОКУМЕНТА

`<html>` ← Начало документа

`<head>` ← Начало заголовка

*Здесь размещается служебная информация.  
Пользователь ее не видит.*

`</head>` ← Конец заголовка

`<body>` ← Начало тела документа

*Здесь размещается содержание документа.  
Именно это видит пользователь.*

`</body>` ← Конец тела документа

`</html>` ← Конец документа

## *Пример HTML-страницы:*

```
<!DOCTYPE html>
```

```
<HTML>
```

```
  <HEAD>
```

```
    <TITLE>Моя первая страница</TITLE>
```

```
  </HEAD>
```

```
<BODY>
```

Простейший пример HTML-страницы, содержащий обязательные теги

```
</BODY>
```

```
</HTML>
```



Простейший пример HTML-страницы, содержащий обязательные теги



## *Пример цветной HTML-страницы:*

```
<!DOCTYPE html>
```

```
<HTML>
```

```
  <HEAD>
```

```
    <TITLE>Моя первая страница</TITLE>
```

```
  </HEAD>
```

```
<BODY  BGCOLOR="#505050" TEXT="blue">
```

Простейший пример HTML-страницы, содержащий обязательные теги с отображением голубого текста на темно-синем фоне

```
</BODY>
```

```
</HTML>
```



Простейший пример HTML-страницы, содержащий обязательные теги с отображением голубого текста на темно синем фоне

# Комментарии

Комментарии открываются символьной последовательностью, начинающейся с открывающей угловой скобки, восклицательного знака и двух знаков тире

<!-- закрываются сочетанием -->.

**Формат комментария:**

<!-- текст комментария -->

# Цветовые спецификации

Определить тот или иной цвет при создании web-страницы можно двумя методами: посредством задания специальных *цветовых меток*, обозначающий названия цветов словами, или **с помощью шестнадцатеричного цифрового кода**, заменяющего эти метки.

Для вывода цветных изображений на экран монитора применяется так называемая палитра RGB (Red, Green, Blue). Согласно данному стандарту, теоретические основы которого были разработаны ученым-физиком Г.Гельмгольцем, можно получить абсолютно любой желаемый оттенок путем смешивания всего лишь трех красок: красной, зеленой и синей.

# Цветовые спецификации

Для обозначения цвета шестнадцатеричным цифровым кодом перед самим кодом ставится символ `<#>`, далее следует набор из шести знаков в шестнадцатеричной системе счисления.

Цвет	Символьная метка	Цифровой код
Белый	White	#FFFFFF
Черный	Black	#000000
Зеленый	Green	#008000
Светло-зеленый	Lime	#00FF00
Серый	Gray	#808080
Светло-серый	Silver	#C0C0C0
Желтый	Yellow	#FFFF00
Темно-бордовый	Maroon	#800000
Синий	Blue	#0000FF
Темно-синий	Navy	#000080
Голубой	Aqua	#00FFFF
Изумрудный	Teal	#008080
Красный	Red	#FF0000
Пурпурный	Purple	#800080
Розовый	Fuchsia	#FF00FF
Оливковый	Olive	#808000

Для передачи цветов в html предпочтительнее пользоваться шестнадцатеричным цифровым кодом.



# Форматирование текста

# Escape-последовательности

Иногда возникает необходимость использования в тексте символы, зарезервированные для обозначения элементов кода HTML. Это могут быть угловые скобки, прямые кавычки и т.д. Т.е. необходимо, чтобы символы отображались как текст, интерпретатор же воспринимает их как часть команд разметки гипертекста. Поэтому существуют escape-последовательности, или их еще называют **«нотации символьных объектов»**. Они начинаются символом «&» и заканчиваются «;», а между ними размещается сама команда, записываемая в строчном регистре.

## Escape-последовательности

Символ	Значение	Именной примитив
<	Левая угловая скобка	&lt;
>	Правая угловая скобка	&gt;
“	Прямые кавычки	&quot;
&	Амперсant	&amp;
	Неразрывный пробел	&nbsp;
©	Символ копирайта	&copy;
®	Символ зарегистрированной торговой марки	&reg;
=	Знак равенства	&equiv;
←	Стрелка влево	&larr;
→	Стрелка вправо	&rarr;
↑	Стрелка вверх	&uarr;
↓	Стрелка вниз	&darr;

# Тег стандартного абзаца

`<P align=аргумент>`

Текст, заключенный в абзац

`</P>`

Аргументы ALIGN:

**right** – выравнивание текста по правому краю;

**left** – выравнивание текста по левому краю;

**center** – выравнивание текста по центру;

**justify** – выравнивание текста по ширине.

Если при использовании тега `<P>` не задано никаких атрибутов, по умолчанию используется левое выравнивание.

Для выравнивания текста по центру можно использовать тег:

`<CENTER>`

Текст

`</CENTER>`



# Абзац

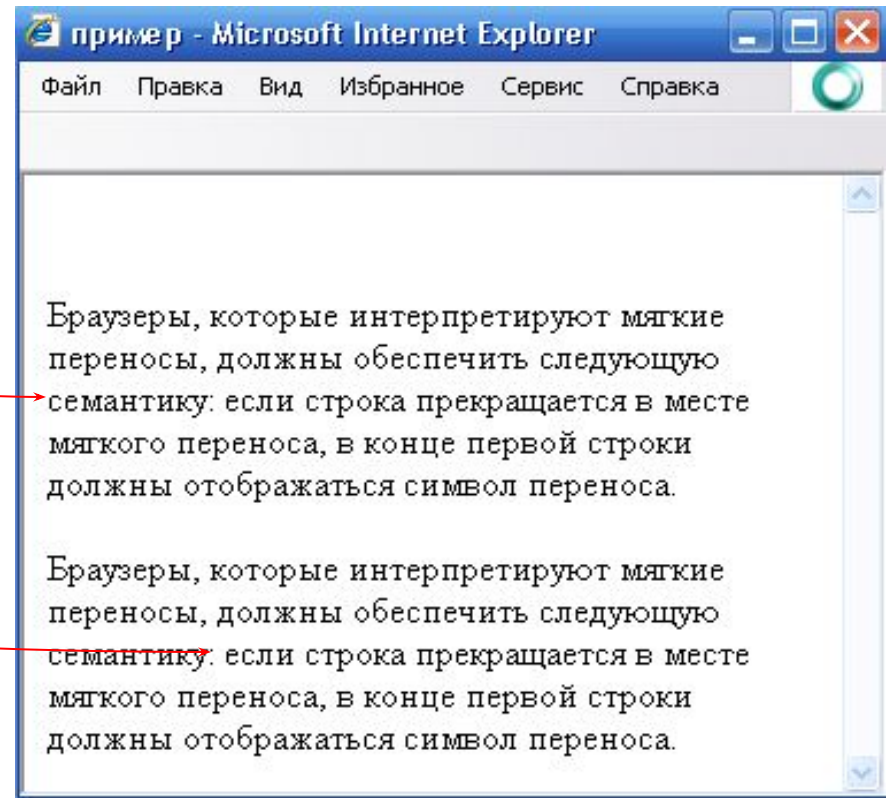
<p>

Браузеры, которые интерпретируют мягкие переносы, должны обеспечить следующую семантику: если строка прекращается в месте мягкого переноса, в конце первой строки должны отображаться символ переноса.

</p>

<p>

Браузеры, которые интерпретируют мягкие переносы, должны обеспечить следующую семантику: если строка прекращается в месте мягкого переноса, в конце первой строки должны отображаться символ переноса.



# Авторское форматирование

<pre>

Я думал,

сердце позабыло

Способность

легкую страдать,

Я говорил:

тому, что было,

Уж не быва

Уж н

Прошли восторги,

и печали,

И легков

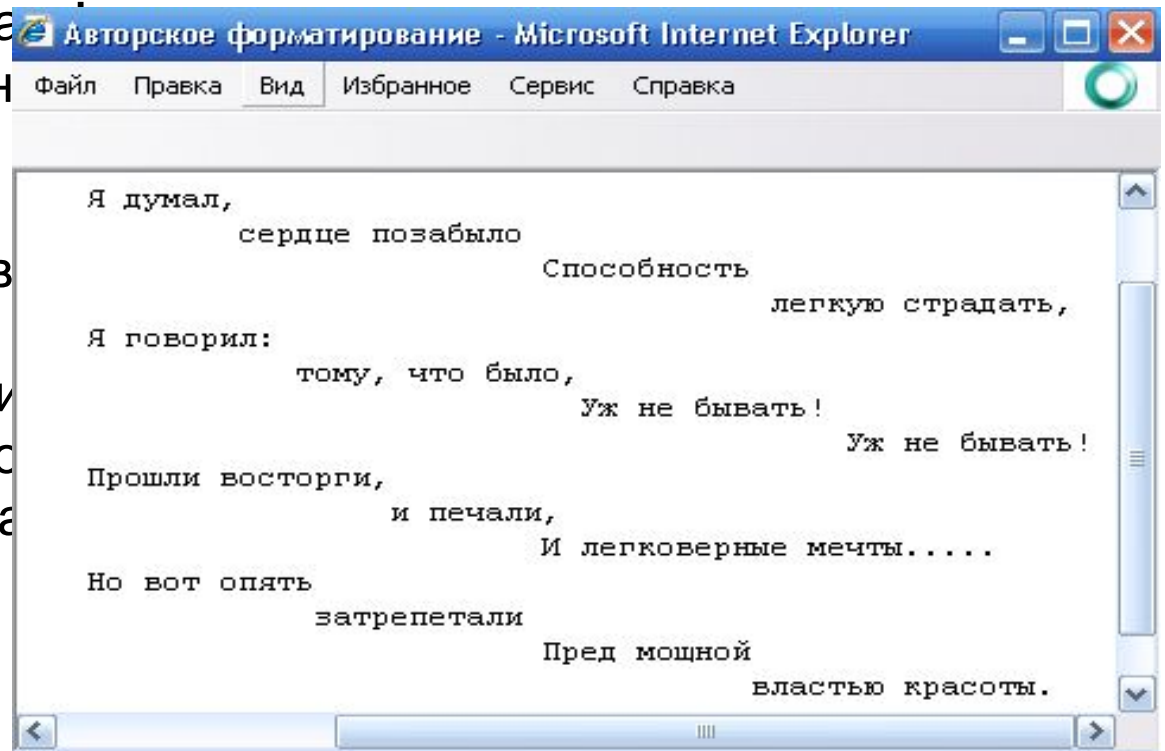
Но вот опять

затрепетали

Пред мощно

вла

</pre>



# Разрывы строк

Для того, чтобы внутри тестового блока браузер отобразил перевод строки, например, для отделения одного абзаца от другого, в конце строки следует поместить тег **<BR>**. Следующее слово, встречающееся за данным тегом, будет перенесено на новую строку. Возможно использование нескольких тегов **<BR>** подряд для задания последовательности из нескольких пустых строк.

**Команда <BR> не имеет закрывающего тега.**

Иногда тег **<BR>** содержит атрибут **clear**, который может принимать одно из трех значений: **left**, **right**, **all**. Указанные параметры позволяют переносить текст со смещением новой строки соответственно к левой, правой или обеим границам экрана одновременно. Синтаксис записи этого тега с атрибутом **clear**:

**<BR clear="значение">**

# Разрывы строк

Некоторые браузеры переносят строки самостоятельно, причем разрывают строку в том месте, где она выходит за границы экрана. Если программисту необходимо отобразить текстовую строку фиксированной длины без переносов, используется тег `<NOBR>`:

`<NOBR>` строка без переносов `</NOBR>`

При использовании данного тега в случае, если длина строки превышает ширину экрана, в нижней части окна браузера появляется горизонтальная полоса прокрутки.

# Шрифты

*Элемент* <FONT> определяет свойства шрифта.

Синтаксис записи тега <FONT>:

```
<FONT face="значение1" size="значение2" color="значение3">
```

текст

```
</FONT>
```

# Шрифты

Значением атрибута **face** является название шрифта, который используется для отображения текста, заключенного в теги `<FONT>` `</FONT>`. Если, например, вместо параметра "значение1" в приведенном примере вы подставите "Arial", помеченный тегом `<FONT>`, текст будет отображаться именно этим шрифтом. *В атрибуте **face** в качестве значений допустимо указывать несколько гарнитур шрифтов, разделяя их запятой.* Для отображения текста браузер применит одну из перечисленных гарнитур, имя которой первой совпадает с имеющимся в его распоряжении набором гарнитур.

```
<FONT face="Helvetica, Arial">
```

# Шрифты

Атрибут **size** указывает на размер шрифта. Параметр этого атрибута может быть описан либо абсолютной, либо относительной величиной. Абсолютная величина подразумевает использование в качестве параметра целого числа, указывающего на высоту шрифта в пунктах. Относительная же величина, обозначаемая целым числом со знаком плюс или минус (например, +2 или -1), — это количество пунктов, которые следует прибавить или отнять от размера шрифта, используемого браузером по умолчанию (12pt).

Так, запись `<FONT size="+1">` говорит о том, что размер помеченного таким тегом текста будет на один пункт больше, чем обычный текст документа. Кроме того, размер шрифта можно записывать целыми числами от 1 до 7. В этих единицах основному шрифту, используемому по умолчанию, присваивается размер 3. Самый мелкий шрифт имеет размер 1, самый крупный — 7. Каждый шрифт в шкале «1-7» на 20% больше или меньше размера основного шрифта.

Атрибут **color** определяет цвет шрифта и подставляется либо цифровым кодом, либо символьной меткой.

# Шрифты

Очевидно, что при использовании тега <FONT> любой из его атрибутов может быть опущен.

```
<FONT face="Arial" size="+2" color="#0000FF">
```

отрывок текста 111111111 <BR>

```
</FONT>
```

```
<FONT face="Arial" size=5 color="olive">
```

отрывок текста 2222222222222222 <BR>

```
</FONT>
```

```
<Font size=6> текст 333333333 <BR> </Font>
```

```
<Font size="+2"> текст 444444444 <BR> </Font>
```

```
<Font size=-1> текст 555555555 </Font>
```

Моя первая страница x Моя первая страни

file:///G:/1.html

Mail.Ru

отрывок текста 111111111

отрывок текста 2222222222222222

текст 333333333

текст 444444444

текст 555555555



# Изменение основного шрифта

По умолчанию текстовое содержимое документа отображается основным шрифтом, заданным в начальных установках браузера. Обычно основной шрифт имеет размер 3 и гарнитуру Times New Roman. Для изменения свойств основного шрифта предназначен одиночный тег `<BASEFONT>`, который имеет атрибуты `size`, `color` и `face`.

`<BASEFONT size=5>`

# Задание цвета фона

Цвет фона всего документа определяется атрибутом bgcolor тега <BODY>.

```
<BODY bgcolor="blue">
```

# Цвет текста

Для определения цвета шрифта можно пользоваться одним из 3 способов:

- ❖ назначить цвет символов всего документа с помощью атрибута text тега <BODY>

```
<BODY text="green">
```

- ❖ если в документе определен основной шрифт с помощью тега <BASEFONT>, можно дополнить этот элемент атрибутом color

```
<BASEFONT size=5 color="red">
```

- ❖ для текстовых фрагментов цвет задается с помощью атрибута color в тегах <FONT>, при этом такое задание цвета отменяет действие атрибута text тега <BODY>

```
<FONT color="red">
```

# Элементы стиля шрифтов

**<B>** Выделение текста полужирным шрифтом **</B>**

**<I>** Выделение курсивным шрифтом **</I>**

**<U>** Выделение текста подчеркиванием **</U>**

**<TT>** Моноширинный текст **</TT>**

**<SMALL>** Уменьшение размера шрифта на 10-20% относительно текущего размера шрифта **</SMALL>**

**<BIG>** Увеличение размера шрифта на 10-20% относительно текущего размера шрифта **</BIG>**

**<STRIKE>** Зачеркнутый текст **</STRIKE>**

**<S>** Зачеркнутый текст **</S>**

**<SUB>** Нижний индекс **</SUB>**

**<SUP>** Верхний индекс **</SUP>**

Данные пары тегов можно вкладывать одна в другую. Например, выделить текст жирным курсивным шрифтом можно так:

**<B><I>**Текст, выделенный жирным курсивным шрифтом**</I></B>**

# Фразеологические элементы

*Фразеологический элемент* не указывает точного способа отображения текста, а лишь определяет, что содержимое элемента несет определенную смысловую нагрузку и должно быть каким-то образом выделено относительно обычного текста.

**<EM>** *Элемент акцентирования.* Большинство браузеров воспринимают его как выделение текста курсивом. **</EM>**

**<STRONG>** *Элемент акцентирования.* Этот элемент обычно интерпретируется браузерами как выделение полужирным шрифтом. **</STRONG>**

**<DFN>** *Термин и его определение.* Отображается курсивом. **</DFN>**

**<KBD>** *“Текст, набранный с клавиатуры”.* Отображается шрифтом Courier. **</KBD>**

**<VAR>** *Задание переменной или ее значения.* Содержимое выводится курсивом. **</VAR>**

**<CODE>** *Воспроизведение специальных символов.* Применяется в случаях, когда необходимо показать, что текст является исходным кодом какой-либо программы. **</CODE>**

**<SAMP>** *Воспроизведение буквенных символов.* Указывает, что текст является образцом. **</SAMP>**

**<ACRONYM title=“значение”>** *Элемент аббревиатуры.* В содержимом располагается сокращенное название аббревиатуры. Расшифровка термина появляется в виде всплывающей подсказки, текст которой задается значением атрибута title. Содержимое ACRONUM появляется обычным текстом. Назначение ACRONUM состоит в подготовке информации для поисковых серверов по расшифровке аббревиатур.

**</ACRONYM>**

# Оформление цитат

`<BLOCKQUOTE cite="URL">`

Текст длинной цитаты

`</BLOCKQUOTE>`

*Задаёт отображение длинных цитат отдельным текстовым блоком. Содержимое элемента выводится с небольшим отступом от левого и правого краев страницы (по умолчанию отступ равен 40px). Перед цитатой и после нее браузер оставляет по одной пустой строке. Атрибут `cite` предназначен для указания URL-адреса цитируемого документа. Этот атрибут не влияет на отображение цитаты и носит справочный характер.*

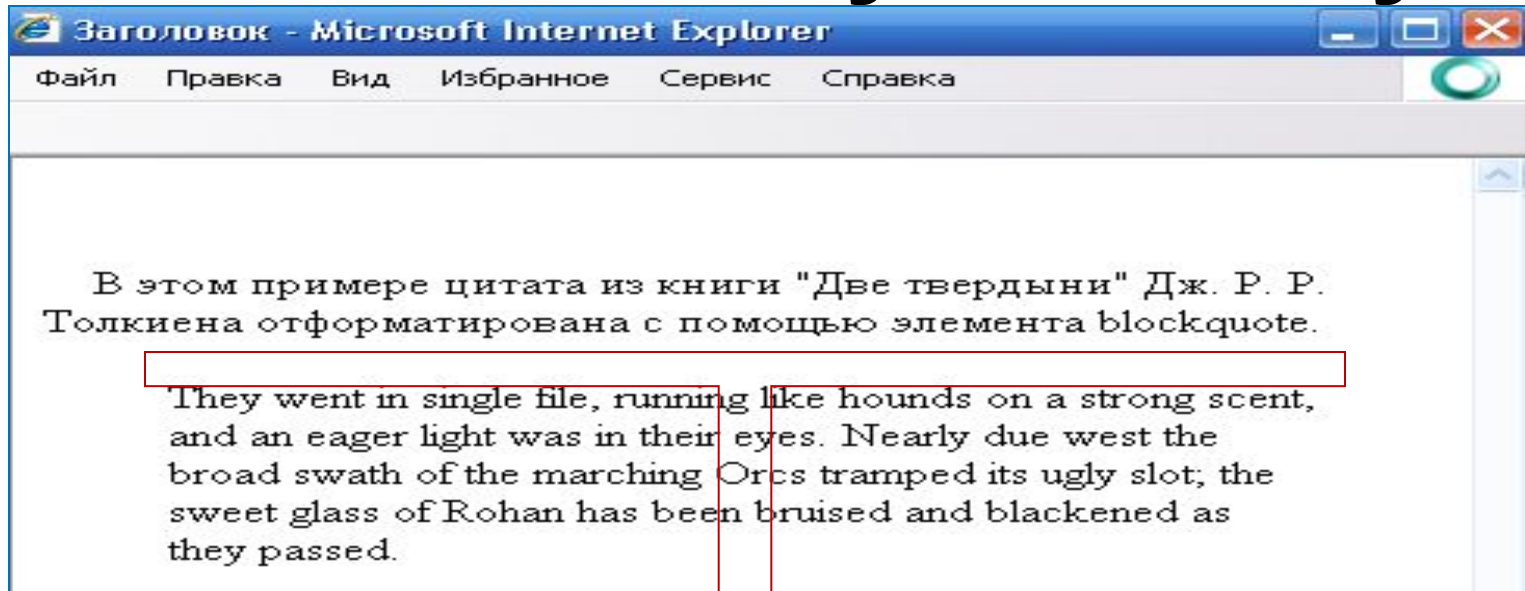
`<Q>` Текст короткой цитаты `</Q>`

*Тег применяется для форматирования коротких цитат. Содержимое элемента отображается без разрыва строки.*

`<CITE>` Текст цитаты `</CITE>`

*Цитаты, оформленные с помощью этого элемента, выделяются курсивным шрифтом.*

# Создаем блочную цитату



`<blockquote>`

They went in single file, running like hounds on a strong scent, and an eager light was in their eyes. Nearly due west the broad swath of the marching Orcs tramped its ugly slot; the sweet glass of Rohan has been bruised and blackened as they passed.

`</blockquote>`

# Элемент <ADDRESS>

Используется для размещения в документе контактной информации. Элемент представляет собой блок, содержимое которого отображается курсивом и внутрь которого можно помещать другие элементы, например, разрывы строк, рисунки, гиперссылки и т.д., но нельзя вставлять блоки (разделы, списки, абзацы).

Формат элемента:

<ADDRESS>

Содержимое

</ADDRESS>

Пример:

<ADDRESS>

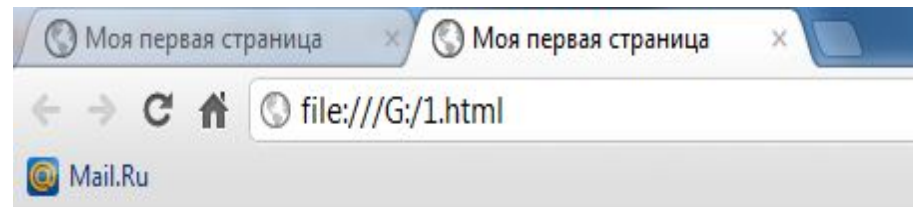
Екатерина Семенова <BR>

Ведущий менеджер

Телефон 123-123-123

<A href="URL">realiz@kdk.com</A>

</ADDRESS>



Екатерина Семенова

Ведущий менеджер Телефон 123-123-123 [realiz@kdk.com](mailto:realiz@kdk.com)



# Атрибуты тега <BODY>

Общий формат записи тега <BODY> со всеми допустимыми атрибутами имеет вид:

```
<BODY background="URL" bgcolor="значение1"  
text="значение2" link="значение3"  
vlink="значение4" alink="значение5">
```

тело документа HTML

```
</BODY>
```

# Атрибуты тега <BODY>

Атрибут **background** позволяет поместить на web-страницу фоновый рисунок, записав в качестве параметра атрибута URL этого рисунка. URL можно задавать либо в виде полного адреса Интернета (например, "http://www.server.ru/images/имя\_файла.gif"), либо в виде сокращенного адреса с указанием пути к директории на текущем сервере, в которой хранится данное изображение (например, "../images/имя\_файла.gif").

Допускается просто указывать имя графического файла, если он хранится в той же директории, что и использующий его файл HTML. Данное изображение может иметь любой размер, поскольку при интерпретации кода оно многократно повторяется, заполняя все доступное пространство в окне браузера.

# Атрибуты тега <BODY>

Атрибут **text** позволяет задать цвет текста для всего документа в целом. Однако он может быть изменен в определенном участке текста путем использования команды <FONT> с атрибутом **color**.

Атрибут **bgcolor** используется для задания фонового цвета всему документу. Атрибуты **bgcolor** и **background** не исключают друг друга, однако у последнего имеется приоритет. Это означает, что в случае, когда заданы оба эти атрибута, сначала выполняется заливка web-страницы цветом, назначенным в атрибуте **bgcolor**, поверх которой размещается изображение, заданное атрибутом **background**.

# Атрибуты тега <BODY>

Атрибут **link** задает цвет, которым отображается непосещенная гиперссылка, то есть ссылка, к которой посетитель данного web-сайта еще не обращался. По умолчанию ей присваивается значение "blue" (#0000FF). В свою очередь, атрибут **vlink** указывает на цвет посещенной ссылки, значение которой по умолчанию — "purple" (#800080).

Атрибут **alink** определяет цвет гиперссылки в момент нажатия. По умолчанию данный атрибут также имеет значение "purple".

# Атрибуты тега <BODY>

Пример использования тега <BODY> со всеми допустимыми атрибутами:

<BODY

background="http://www.myserver.com/images/back.jpg"

bgcolor="#000000" text="#FFFFFF" link="#008000"

vlink="#800080" alink="#FF0000">

Тело документа HTML

</BODY>

# Применение многоязычного текста

Указание языка, на котором составлен документ или его фрагмент выполняется атрибутом **lang**. В качестве значения этого атрибута представляется код языка (**en** – английский, **en-US** – английский США, **de** – немецкий, **ru** – русский, **ua** – украинский).

## Пример:

`<HTML lang="de">` – документ составлен на немецком языке,


`<Q lang="ua">` – цитата на украинском.

# Направление чтения текста

Задается атрибутом **dir**, принимающим значения:

- ❖ **ltr** – слева направо,
- ❖ **rtl** – справа налево.

Атрибут задается для текстов на арабском языке или иврите.



# **Разделы, заголовки и горизонтальные линии**



# Разделы

Формат задания раздела:

`<DIV>`

HTML-код или текст

`</DIV>`

или

`<SPAN>`

HTML-код или текст

`</SPAN>`

Разделы используются для задания параметров текста и его расположения. К разделу можно применять любые атрибуты стиля. Теги `<DIV>` и `<SPAN>` отличаются тем, что при использовании тега `<DIV>` браузер производит перенос раздела на новую строку и перед ним автоматически устанавливается пустая строка. При использовании тега `<SPAN>` переноса не возникает.

# Заголовки

Заголовки позволяют разделить web-страницу на логически законченные блоки и помогают ориентироваться в содержимом документа.

Спецификация HTML позволяет программисту использовать шесть различных стилей, или, как их еще принято называть, логических уровней заголовков.

# Заголовки

Для того чтобы отобразить заголовок на web-странице, необходимо использовать тег `<Hn>`, где  $n$  — целое число от 1 до 6, обозначающее номер уровня заголовка, причем самым верхним уровнем принято считать 1. Синтаксис записи этой команды:

```
<Hn align="параметр">
```

Текст заголовка

```
</Hn>
```

Атрибут **align** дает возможность web-мастеру определить расположение заголовка в окне браузера. Параметр этого атрибута может принимать одно из трех возможных значений:

- **right** — позиционирование заголовка по правой границе документа;
- **left** — позиционирование заголовка по левой границе документа;
- **center** — позиционирование заголовка по центру документа.

# Заголовки

<h1>Заголовок 1</h1>

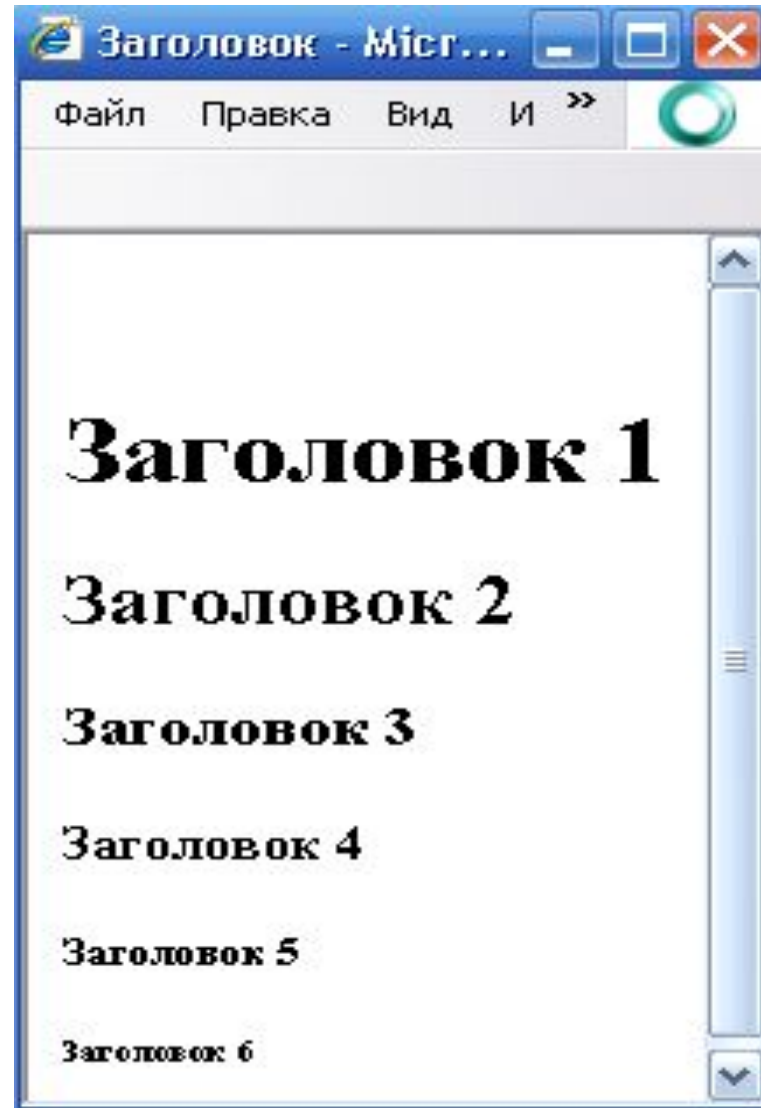
<h2>Заголовок 2</h2>

<h3>Заголовок 3</h3>

<h4>Заголовок 4</h4>

<h5>Заголовок 5</h5>

<h6>Заголовок 6</h6>



# Заголовки

По умолчанию заголовки выравниваются по левому краю страницы.

Заголовки как блочные элементы отделяются от текста пустыми строками.

Пример:

```
<H1 align="center">
```

ДОБРО ПОЖАЛОВАТЬ НА МОЮ СТРАНИЧКУ!

```
</H1>
```

# Горизонтальные линии

Чтобы визуально отделить часть объектов документа от других, применяются горизонтальные линии. Горизонтальная линия – это графический элемент, имеющий длину, толщину и цвет. Для создания таких линий в HTML используется непарный тег `<HR>` (сокращение от Horizontal Ruler). В спецификации HTML для отображения вертикальных линий не предусмотрено специальных тегов.

Формат создание стандартной линии (по умолчанию стандартная линия занимает всю ширину окна браузера, а её толщина составляет 2px, верхняя часть линии несколько темнее нижней):

`<HR>`

Формат сплошной темной линии без тени:

`<HR noshade>`

# Горизонтальные линии

Формат горизонтальной линии с указанием свойств:

```
<HR align="значение" width="значение" size="значение" color="значение">
```

Атрибут **align** задает выравнивание линии и может принимать значения **center**, **left** или **right**. По умолчанию линии выравниваются по центру страницы.

Атрибут **width** задает длину линии в пикселях или процентах.

Например, тег `<HR width=25>` определяет горизонтальную линию шириной 25 пикселей.

Если значение атрибута задается в процентах (ставится символ %), то ширина линии вычисляется относительно ширины окна, например `<HR width="25%">`. Задание длины линии в относительных единицах предпочтительнее.

Толщина линии задается атрибутом **size**. Значение этого атрибута задается в пикселях в диапазоне от 1 до 175. Присвоение больших значений игнорируется. По умолчанию **size=2px**.

Цвет линии определяется атрибутом **color**.