



# ОСНОВЫ HTML

Учебное пособие для уроков  
информатики

# Содержание

Урок 1. Общие представления о языке HTML HTML – слайды 3-12

Урок 2. Ввод и форматирование текста Урок 2. Ввод и форматирование текста WebУрок 2. Ввод и форматирование текста Web-страницы – слайды 13-22

Урок 3. Специальный текст и символы – слайды 23-28

Урок 4. Графика на Графика на WebГрафика на Web-страницах – слайды 29-34

Урок 5. Гиперссылки на Гиперссылки на WebГиперссылки на Web-страницах – слайды 35-41

Урок 6. Таблицы – слайды 42-49

Приложения –

Литература –



# Урок 1

## Общие представления о языке HTML

# Общие понятия HTML

HTML (Hyper Text Markup Language – язык гипертекстовой разметки) является языком программирования, предназначенным для разработки Web-страниц.

Язык HTML представляет собой коллекцию управляющих символов – *дескрипторов*. Для настройки внешнего вида и функционирования элемента страницы устанавливаются его *атрибуты*.

Обозреватели распознают дескрипторы языка HTML и преобразуют код документа в Web-страницу. Приложения обозревателей разрабатываются с учетом мировых стандартов языка HTML, поэтому Web-страницы выглядят одинаково в окнах разных обозревателей.

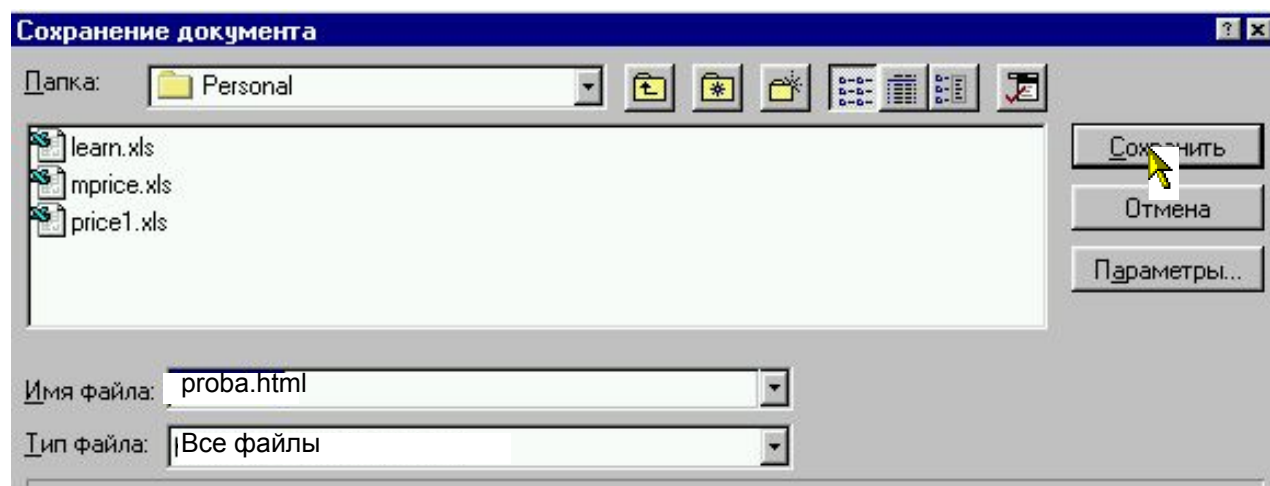
# Историческая справка

Сеть Word Wide Web (WWW) родилась в 1989 году в Женеве, Швейцария. Придумал мировую паутину Тим Бернес-Ли. Создание WWW не было для Тима внезапным озарением. Еще в 1980 году им была разработана для личного использования система маркирования гипертекста Enquire – ранний прототип HTML.

У HTML был еще более ранний предшественник – GML (Generalized Markup Language – язык обобщенной разметки), разработанный Чарльзом Гольдфардом в 1969 году. В 1986 году этому языку, переименованному в SGML (стандартный GML), был присвоен статус международного стандарта. Язык HTML создавался Тимом Бернес-Ли в соответствии со стандартами SGML. Но идея гиперссылок и URL-адресов в стиле структуры доменных имен [www. имя. имя.](#) была впервые реализована именно Тимом Бернес-Ли.

С тех пор язык HTML много раз был модифицирован, особенно в 90-тые годы, когда Интернет развивался наиболее динамично.

# Создание HTML-документа



Для создания HTML-документа подойдет любой текстовый редактор (например, Блокнот). Введите в текстовый файл HTML-код, а затем сохраните этот файл с расширением **.html**.

Приложения-обозреватели автоматически преобразуют код HTML в Web-страницу с текстом, графикой и встроенными объектами мультимедиа.

# Просмотр и изменение HTML-кода

Чтобы отобразить исходный код Web-страницы, нужно в контекстном меню выбрать команду **Открыть с помощью...-> Блокнот** или **Просмотр HTML-кода**.

Если вы хотите изменить HTML-код, внесите необходимые изменения в исходный HTML-документ, сохраните его (**Файл->Сохранить**), затем обновите Web-страницу (**Вид->Обновить** или нажмите клавишу **F5** или значок на панели инструментов)



# Обязательные дескрипторы HTML

`<HTML>`

`<HEAD>`

*параметры страницы*

`</HEAD>`

`<BODY>`

*текст страницы*

`</BODY>`

`</HTML>`

`< HTML>...</ HTML>` - устанавливаются в начале и конце документа.

`<HEAD>... </HEAD>` - заголовок Web-страницы.

`<BODY>... </BODY>` - основной раздел Web-страницы.

Дескрипторы бывают парные и непарные. Обязательные дескрипторы – парные: открывающие (`< HTML>`) и закрывающие (`</ HTML>`). Отсутствие закрывающего дескриптора может вызвать ошибку при просмотре Web-страницы.



# Дескрипторы раздела заголовка

Перечислим некоторые наиболее часто используемые дескрипторы, устанавливаемые в заголовке (между тегами `<HEAD>... </HEAD>`)

`<TITLE>... </TITLE>` - заголовок Web-страницы.

`<META name='параметр' content='значение'>` - непарный дескриптор для установки метапараметров (имя автора, описание Web-страницы и ключевые слова для поисковиков). Этот дескриптор мы рассмотрим позднее.

`<SCRIPT>...</SCRIPT>` - код сценария, выполняемого во время загрузки Web-страницы. Об этом дескрипторе мы также поговорим позднее.

# Дескрипторы основного раздела

В основном разделе Web-страницы содержится текст документа с дескрипторами, которые выполняют форматирование текста и добавление элементов Web-страницы. Эти дескрипторы также бывают как парные (например, `<P>...</P>`), так и непарные (`<BR>`)

Более подробно об этом разделе мы поговорим на следующих уроках.

# Практика


```
<HTML>
<HEAD>
<TITLE>Проба</TITLE>
</HEAD>
<BODY>
Я хочу сделать Web-страницу
</BODY>
</HTML>
```

Откройте Блокнот (**Пуск ->Все программы ->Стандартные ->Блокнот**) и сохраните документ с расширением **.html** (**Файл ->Сохранить как...** ->укажите тип файла **Все файлы** ->имя документа **proba. html**)

Введите HTML-код из примера.

Сохраните документ (**Файл ->Сохранить**) и закройте его.

Найдите сохраненную Web-страницу, откройте ее, обновите и посмотрите, что у вас получилось.



Откройте HTML-код (правой кнопкой по Web-странице вызовите контекстное меню и выберите команду **Просмотр HTML-кода**).

Внесите изменения в основной раздел Web-страницы (измените текст между дескрипторами **<BODY>... </BODY>**).

Сохраните изменения (**Файл -> Сохранить**) и обновите Web-страницу.

**Поздравляю с успешным началом!**



# Урок 2

## Ввод и форматирование текста Web-страницы

# Разметка текста на странице

Обозреватели Web-страниц игнорируют все отступы, пробелы и разрывы строк, заданные в текстовом редакторе. Абзацы, введенные в Блокноте, объединяются в сплошной текст в окне обозревателя. Поэтому, чтобы разметить текст Web-страницы, следует воспользоваться специальными дескрипторами HTML.



# Создание нового абзаца

Для создания абзаца используется пара блочных дескрипторов `<P>...</P>` в основной раздел Web-страницы (между дескрипторами `<BODY>... </BODY>`).

По умолчанию текст выравнивается по левому краю, слова автоматически переносятся, когда текст достигает края окна обозревателя. Изменить выравнивание можно с помощью атрибута `ALIGN` в дескрипторе `<P>`.

Дескриптор `<NORB>... </NORB>`, вставленный между тегами `<P>...</P>` блокирует автоматический перенос слов.

`<P ALIGN="left">Выравнивание по левому краю</P>`

`<P ALIGN="center">Выравнивание по центру</P>`

`<P ALIGN="right">Выравнивание по правому краю</P>`

`<P ALIGN="justify">Выравнивание по ширине</P>`

`<P> <NORB>Этот текст не будет переноситься на другую строку, даже если его длина больше ширины окна</NORB> </P>`

*Вставьте листинг в основной раздел вашей Web-страницы и посмотрите, что у вас получилось.*  
На содержание

# Разрыв строки

Если текст нужно начать с новой строки, используется непарный дескриптор `<BR>`. Обозреватель покажет текст с новой строки без добавления пустой строки. Дескриптор `<BR>` часто используют для добавления рисунка с новой строки, так как по умолчанию обозреватель отображает рисунок в той же строке, где находится дескриптор `<IMG>`.

## Пробелы и отступы текста

Если вы хотите сохранить форматирование текста из Блокнота, воспользуйтесь парой `<PRE>...</PRE>`. Эти дескрипторы заставляют обозреватель сохранять все пробелы и табуляции в тексте, заключенном между ними.

Комбинация символов `&nbsp;` устанавливает неразрывный пробел. Символы, между которыми стоит неразрывный пробел, не будут переноситься на другую строку в окне обозревателя.

Для визуального выделения абзаца можно воспользоваться дескрипторами `<BLOCKQUOTE>...</BLOCKQUOTE>`, которые автоматически задают отступ от левого и правого краев окна.



# Практика

Задание 1. Введите в документ [proba.html](#) между дескрипторами `<P>...</P>` строфу из стихотворения «Зеленый шум»:

Люби, покуда любится,  
Терпи, покуда терпится,  
Прощай, пока прощается,  
И - Бог тебе судья!

Н. А. Некрасов

Проверьте, что у вас получилось.

Задание 2. Вставьте в конце каждой строки дескриптор `<BR>`. Проверьте, как изменился вид документа.

Задание 3. Вставьте перед именем автора три символа `&nbsp;`. Посмотрите, как теперь выглядит текст. Добавьте еще необходимое количество символов.

# Текстовые заголовки

`<H1>Заголовок первого уровня</H1>`  
`<H2>Заголовок второго уровня</H2>`  
`<H3>Заголовок третьего уровня</H3>`  
`<H4>Заголовок четвертого уровня</H4>`  
`<H5>Заголовок пятого уровня</H5>`  
`<H6>Заголовок шестого уровня</H6>`

*Вставьте листинг в вашу Web-страницу и посмотрите, что у вас получилось.*

Заголовки шести уровней устанавливаются с помощью парных дескрипторов от `<H1>...</H1>` до `<H6>...</H6>`, причем размер заголовка уменьшается с каждым уровнем. Дескрипторы заголовков устанавливаются в «тело» Web-страницы и имеют больший приоритет для поисковых программ, таких как Yahoo, Rambler или Яндекс. Поэтому, чем четче заголовок будет выражать суть Web-страницы, тем больше вероятность того, что ваш документ будет найден в Интернете.

# Установка шрифта

Дескриптор `<FONT>...</FONT>` позволяет устанавливать параметры текста: шрифт, его размер и цвет.

Шрифт выбирается с помощью параметра `FACE`. В значении параметра следует указать имя шрифта или нескольких шрифтов. Например:

`<FONT FACE='Tahoma, Arial, Georgia'> текст </FONT>`

Число шрифтов в списке не ограничено, но некоторые из них могут отсутствовать на компьютере пользователя. Поэтому обязательно указывайте распространенные шрифты. Обозреватель будет применять шрифты в том порядке, как они записаны.

Размер шрифта устанавливается с помощью параметра `SIZE`, значение которого меняется от 1 до 7. Например:

`<FONT SIZE=1> текст </FONT>`

Шрифт увеличивается при увеличении размера.

# Цвет текста и фона

Цвет	Шестнадцатеричный код	Именной код
Белый	#FFFFFF	White
Бирюзовый	#008080	Teal
Желтый	#FFFF00	Yellow
Зеленый	#008000	Green
Золотой	#FFD700	Gold
Красно-коричневый	#800000	Maroon
Красный	#FF0000	Red
Лимонный	#00FF00	Lime
Морской волны	#00FFFF	Aqua
Оливковый	#808000	Olive
Пурпурный	#800080	Purple
Серебряный	#C0C0C0	Silver
Серый	#808080	Gray
Синий	#0000FF	Blue
Темно-голубой	#000080	Navy
Фуксиновый	#FF00FF	Fuchsia
Черный	#000000	Black

Цвет текста можно установить двумя способами:

1. с помощью параметра **COLOR** в дескрипторе **<FONT>**;
2. с помощью параметра **TEXT** в дескрипторе **<BODY>**.

Значение цвета в обоих случаях устанавливается одинаково: либо с использованием шестнадцатеричного кода, либо с помощью именного кода.

Цвет фона устанавливает параметр **BGCOLOR** в дескрипторе **<BODY>**

# Форматирование текста

Для форматирования текста используются следующие дескрипторы:

`<B>...</B>` - полужирный шрифт;

`<I>...</I>` - курсив;

`<U>...</U>` - подчеркивание;

`<PRE></PRE>` - сохраняет форматирование исходного текста;

`<SUP>...</SUP>` - надстрочный символ;

`<SUB>...</SUB>` - подстрочный символ.

Текст для форматирования помещается непосредственно между дескрипторами пары. Например, создадим запись:  $x^2-3x+2=0$ , **корни этого уравнения** –  $x_1=1$ ,  $x_2=2$ .

`<I>x<SUB>2</SUB>-3x+2=0,</I><B> корни этого уравнения –</B><I>x<SUP>1</SUP>=1, x<SUP>2</SUP>=2</I>`

*Введите листинг в вашу Web-страницу. Удобнее сначала набрать весь текст, а затем расставьте необходимые дескрипторы.*

На содержание

# Практика

Задание 1. Введите в документ [proba.html](#) в дескриптор `<BODY>` параметры `BGCOLOR='yellow' TEXT='blue'`

`<BODY BGCOLOR='yellow' TEXT='blue' >` текст страницы `</BODY>`

Измените значения параметров, пользуясь таблицей цветов. Посмотрите, как изменится вид Web-страницы.

Задание 2. Введите в документ [proba.html](#) между дескрипторами `<P>...</P>` листинг:

```
<FONT SIZE=1 FASE='arial' COLOR='red'> Шрифт 1</FONT> <BR>
<FONT SIZE=2 FASE='arial' COLOR='red'> Шрифт 2</FONT> <BR>
<FONT SIZE=3 FASE='arial' COLOR='red'> Шрифт 3</FONT> <BR>
<FONT SIZE=4 FASE='arial' COLOR='red'> Шрифт 4</FONT> <BR>
<FONT SIZE=5 FASE='arial' COLOR='red'> Шрифт 5</FONT> <BR>
<FONT SIZE=6 FASE='arial' COLOR='red'> Шрифт 6</FONT> <BR>
<FONT SIZE=7 FASE='arial' COLOR='red'> Шрифт 7</FONT> <BR>
```

Измените значения параметров. Посмотрите, как изменится вид Web-страницы.



# Урок 3

## Специальный текст и СИМВОЛЫ

# Нумерованный список

Для создания нумерованного списка используются парные теги `<OL>...</OL>`, задающие начало и конец списка и `<LI>...</LI>`, устанавливающие начало и конец каждого пункта.

TYPE	Тип нумерации
'1'	1, 2, 3, 4, .....(задан по умолчанию)
'i'	i, ii, iii, iv ....
'I'	I, II, III, IV....
'a'	a, b, c, d, ....
'A'	A, B, C, D,.....

Способ нумерации устанавливается с помощью атрибута **TYPE** в дескрипторе `<OL>...</OL>`

Начальный номер пунктов устанавливается с помощью атрибута **START** в дескрипторе `<OL>...</OL>`

Если нужно пропустить часть номеров и продолжить нумерацию с другого номера, то это можно сделать с помощью атрибута **VALUE** в дескрипторе `<LI>...</LI>`, присвоив ему нужный номер



# Маркированный список

<H2>маркированный  
список</H2>

<UL TYPE = 'circle'>

<LI> окружность</LI>

<LI TYPE = 'disk'>круг</LI>

<LI TYPE = 'square'>квадрат</LI>

</UL>

Для создания маркированного списка вместо тегов <OL>...</OL> используются теги <UL>...</UL>, а каждая строка списка - также тегами <LI>...</LI>.

Тип маркера устанавливается значением атрибута **TYPE**, который может принимать следующие значения:

TYPE	Тип маркера
'disk'	●
'circle'	○
'square'	■

# Многоуровневый список

Многоуровневый список создать легко: нужно просто вложить список внутри другого. Попробуйте сделать это самостоятельно.

## Создание отступов

Отступ от левого края создается с помощью следующих тегов:

`<DL>...</DL>` начало и окончание списка определений;

`<DT>...</DT>` заголовок термина;

`<DD>...</DD>` определение термина;

`<DD>&nbsp;</DD>` пустая строка;

`<P><DD> текст абзаца</DD></P>` красная строка

# Бегущая строка

Бегущая строка создается с помощью парного дескриптора `<MARQUEE>...</MARQUEE>`. Текст, расположенный между этими тегами, можно форматировать так же, как обычный текст.

Атрибуты этого дескриптора:

**BEHAVIOR** способ выполнения эффекта бегущей строки

- scroll текст исчезает за краем страницы;
- slide текст остается у левого края страницы;
- Alternate направление перемещения текста меняется на противоположное;

**BGCOLOR** цвет фона бегущей строки;

**DIRECTION** направление перемещения текста;

- left влево;
- right вправо;

**HEIGHT** высота бегущей строки;

**WIDTH** ширина бегущей строки;

**HSPACE** отступ от полей страницы;

**LOOP** число показов текста

- любое целое число, указывающее число повторов;
- Infinite бесконечное повторение.

# Комментарии для разработчиков

Код Web-страницы может быть достаточно сложным, и со временем разработчик может забыть назначение отдельных блоков. Кроме этого, без комментариев достаточно сложно разобраться в коде постороннему человеку. Поэтому рекомендуется помечать блоки кодов специальными командными символами `<!-- комментарий-->`. Текст, расположенный внутри комментария, не отображается в окне обозревателя и не меняет внешний вид страницы.

`<HTML>`

`<HEAD>параметры страницы</HEAD>` `<!-- заголовок Web-страницы -->`

`<BODY>текст страницы</BODY>` `<!-- основная часть Web-страницы -->`

`</HTML>`



# Урок 4

## Графика на Web-страницах

# Форматы графических файлов

В Интернете используется всего три формата графических файлов:

JPEG – для высококачественных полноцветных изображений, фотографий \*.jpg;

GIF – для рисунков, содержащих текст, линии, фигуры и одноцветные области. В этом формате также сохраняются кнопки, логотипы, анимированные рисунки.

PNG – альтернатива двум первым форматам. Этот формат можно использовать во всех вышеперечисленных случаях, но он поддерживается не всеми обозревателями.

# Добавление рисунка на Web-страницу

Чтобы добавить рисунок, используется непарный тег `<IMG SRC='имя файла'>`. Атрибуту `SRC` присваивается путь к файлу на диске или URL-адрес файла рисунка в Интернете. Если рисунок находится в другой папке, то в ссылке указывается также имя папки.

По умолчанию рисунок отображается без границ. При желании границы можно сделать видимыми с помощью атрибута `BORDER`, присвоив ему значение толщины границы в пикселях. `BORDER='0'` – границы нет.

Рисунки замедляют выгрузку Web-страницы, поэтому существуют специальные приемы, занимающие внимание пользователя на время выгрузки. Например, можно создать две копии рисунка: с высоким и низким разрешением и использовать атрибут `LOWSRC`: `<IMG SRC='имя файла с высоким разрешением' LOWSRC='имя файла с низким разрешением' WIDTH='100' HEIGHT='75'>`. При загрузке сначала появится файл с низким разрешением, который потом заменяется файлом с высоким разрешением. Атрибут `ALT` может снабдить рисунок альтернативным текстом, появляющимся вместо незагрузившегося рисунка.

# Размещение рисунка

Размер изображения задается атрибутами **WIDTH** и **HEIGHT** в дескрипторе **<IMG>**. Значения этих атрибутов могут быть указаны в пикселях или в процентах от истинного размера рисунка. Можно одному из изображений задать значение **auto**, чтобы пропорции рисунка сохранялись.

Выравнивание рисунка относительно текста задает атрибут **ALIGN**, принимающий значения:

- **left** – рисунок слева от текста;
- **right** – рисунок справа от текста;
- **top** – текст выровнен по верхнему краю рисунка;
- **middle** – текст выровнен посередине;
- **bottom** – текст выровнен по нижнему краю рисунка.

Отступ текста от рисунка управляется атрибутами **HSPASE** и **VSPASE** в пикселях – слева и справа, над и под рисунком. Все атрибуты устанавливаются в дескрипторе **<IMG>**

На содержание



# Фоновый рисунок



myimg.gif

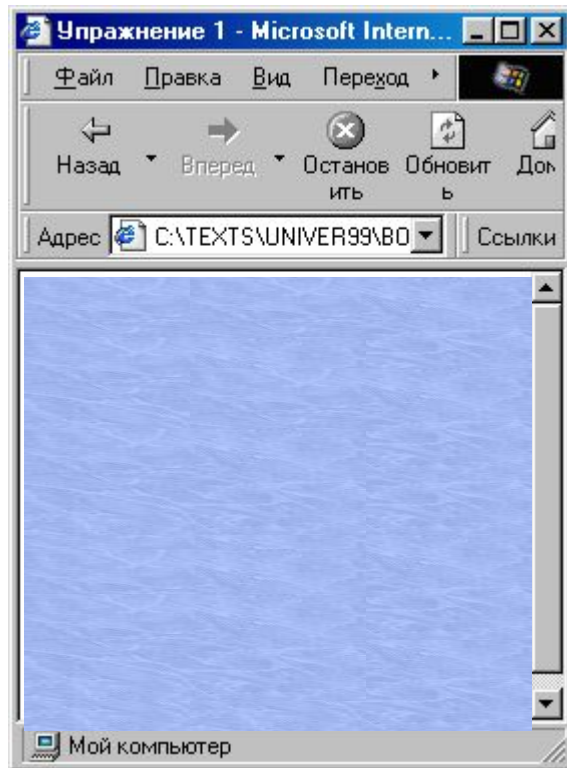


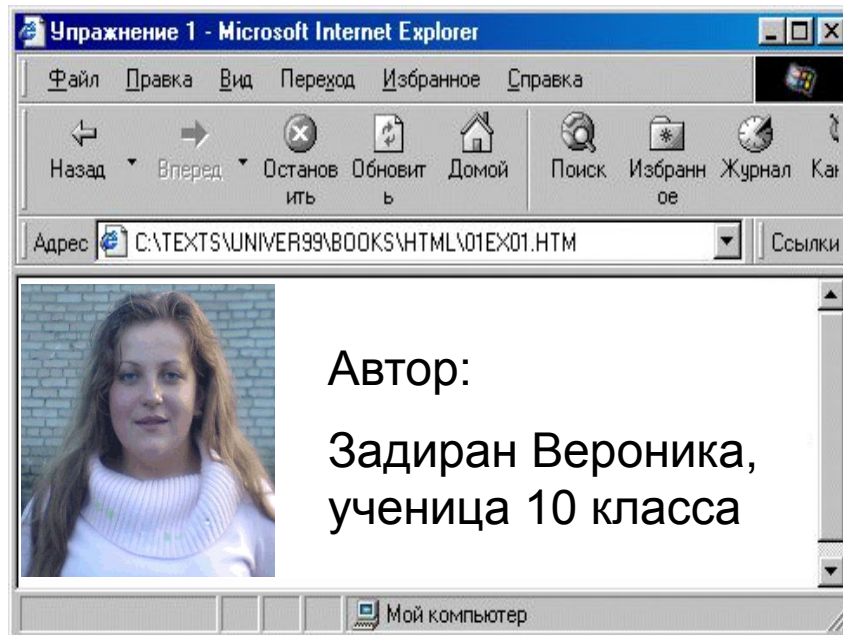
Рисунок можно использовать в качестве фона, установив в дескрипторе **<BODY>** атрибут **BACKGROUND**, указав в его значении адрес расположения рисунка.

**<BODY BACKGROUND='myimg.gif' >**

При этом выбранный рисунок замостит область окна обозревателя.

Чтобы зафиксировать фон (не давать ему прокручиваться), можно установить атрибут **BGPROPERTIES='fixed'** в дескрипторе **<BODY>**

# Практика



1. Создайте авторскую страницу (см. пример).
2. Измените фон страницы
3. Измените выравнивание фотографии относительно текста.
4. Улучшите внешний вид страницы, добавив рисунки, анимацию, горизонтальные и вертикальные линии, рамки и т.п.
5. Сохраните web-страничку, дав ей имя (например, avtor.html)

Вот и готова твоя первая авторская страница!



# Урок 5

## Гиперссылки на Web-страницах

# Создание гиперссылки

`<A HREF='адрес гиперссылки'>гиперссылка</A>`

Гиперссылки создаются с помощью тега `<A>...</A>`.

Атрибуту **HREF** присваиваются следующие значения:

<code>&lt;A HREF='mydoc.html'&gt; мой документ &lt;/A&gt;</code>	адрес URL на другой Web-странице или путь на диске к другому документу;
<code>&lt;A HREF='http://www.narod.ru/'&gt; адрес в Интернете &lt;/A&gt;</code>	адрес URL на ресурс в Интернете
<code>&lt;A HREF='ftp://...../'&gt; адрес на сервере &lt;/A&gt;</code>	адрес URL на документ FTP-сервера
<code>&lt;A NAME='mark1'&gt; текст закладки &lt;/A&gt;</code> <code>&lt;A HREF='mark1'&gt; гиперссылка &lt;/A&gt;</code>	закладка на текущей Web-странице
<code>&lt;A HREF='mailto: myadres@mail.ru'&gt; моя почта &lt;/A&gt;</code>	адрес электронной почты
<code>&lt;A HREF='javascript: open_links();'&gt; адрес гиперссылки &lt;/A&gt;</code>	сценарий на языке Web-программирования

# Оформление гиперссылки

Цвет ссылки нужно прописывать:

- **link** - цвет ссылки,
- **alink** - цвет активной ссылки (нажатой),
- **vlink** - цвет уже посещенной ссылки.

В нашем примере цвета одинаковые, но они могут быть разными, также не карается законом, если цвет ссылки такой же как цвет текста документа (ссылка же выделяется чертой).

```
<body text="#336699" bgcolor="#000000" link="#339999" alink="#339999" vlink="#339999">
```

# Распределение ссылок по картинке

```
<IMG SRC="url"
USEMAP="url#map_name">
```

```
<MAP NAME="map_name"> <AREA
SHAPE=" shape " COORDS="x,y,..."
HREF=" adress" TITLE='подсказка'>
</MAP>
```

Можно создать графическое меню из одной большой картинки таким образом, чтобы каждый элемент рисунка являлся ссылкой. Распределение ссылок по картинке описывается в тэге IMG.

**USEMAP** задает имя элемента рисунка **map\_name**.

**<AREA>** определяет область на картинке посредством параметров **SHAPE** и **COORDS**.

**SHAPE** – форма области.

- **default** -- стандартная форма
- **rect** -- прямоугольник
- **circle** -- круг
- **poly** -- многоугольник произвольной формы

**COORDS** – координаты области. Задаются в пикселах. Круг имеет три координаты (центр и радиус), прямоугольник – четыре (координаты противоположных вершин), а для многоугольника вы должны описать каждую его вершину в двух координатах.

**HREF="адрес"** – определяет ссылку на схеме

# Вставка звукового файла

Для того, чтобы вставить в вашу страничку звуковой файл, например, midi-файл, используйте следующую конструкцию:

```
<EMBED SRC="bob1.mid" WIDTH="140" HEIGHT="50" ALIGN="MIDDLE" BORDER="0" AUTOSTART=TRUE>.
```

В этом тэге были использованы следующие параметры:

**WIDTH** – параметр, определяющий ширину midi-плеера.

**HEIGHT** – параметр, определяющий высоту midi-плеера.

**BORDER** – ширина рамки midi-плеера.

**AUTOSTART** – запустить midi-плеер сразу после того, как загрузится документ HTML.

Точно также загружаются и видеоролики.

# Ссылка в новом окне

По умолчанию, файл по гиперссылке открывается в том же окне, что не всегда удобно. Как же сделать так, чтобы картинка (или любой другой файл-документ) открылась в новом окне? Для этого у тэга `<a></a>` есть параметр `target`

```
<a href="big.jpg" target="_blank">  </a>
```

Значение параметра `target="blank"` - указывает на то, что документ (картинка в нашем случае), на который ведет ссылка, откроется в новом окне браузера



# Практика

1. Вставьте в документ [proba.html](#) ссылку на авторскую страницу:

`<A HREF='avtor.html'>имя автора</A>`

Проверьте, как она работает.

2. Подберите рисунок машины и отдельных ее частей (или растения, или человека и его органов, или план местности и соответствующие фотографии), сохраните их в отдельной папке. Создайте Web-документ, вставьте в него основной рисунок и, используя его как графическое меню, внесите ссылки на остальные рисунки или текст.



# Урок 6

## Таблицы

# Простейшая таблица

```
<table border="1">
```

```
<tr ><!--1 строка таблицы из одной ячейки-->
```

```
    <td >фамилия</td>
```

```
</tr>
```

```
<tr ><!--2 строка таблицы из двух ячеек-->
```

```
    <td > имя      </td>
```

```
    <td > отчество </td>
```

```
</tr>
```

```
</table>
```

Таблица создается с помощью пары дескрипторов `<TABLE>...</TABLE>`, строки и ячейки которой добавляются тегами:

`<TR>...</TR>` - строка таблицы;

`<TD>...</TD>` - ячейка таблицы;

`<TH>...</TH>` - ячейка заголовка.

Сначала задается строка таблицы, затем в ней указывается количество ячеек и их содержимое. Потом аналогично задается следующая строка и так далее. Количество ячеек в строках может быть как одинаковым, так и различным, как в приведенном примере

# Форматирование текста в ячейке

Для выравнивания текста по горизонтали и вертикали используются атрибуты **ALIGN** и **VALIGN**, которым присваиваются следующие значения:

## □ **ALIGN**

- **left** – влево;
- **center** – по центру;
- **right** – вправо.

## □ **VALIGN**

- **top** – вверх;
- **middle** – по центру;
- **bottom** – вниз.

# Форматирование ячеек таблицы

Размеры ячеек устанавливаются с помощью атрибутов **WIDTH** и **HEIGHT**, которым присваиваются значения как в пикселях, так и в процентах от размеров окна браузера.

Для установки цвета ячейки применяется атрибут **BGCOLOR**, принимающий те же значения, что и при установке цвета фона Web-страницы. Также можно применить фоновый рисунок, используя атрибут **BACKGROUND**. Все эти атрибуты можно применять как для форматирования всей таблицы, так и для форматирования ее отдельных ячеек.

Очень часто таблицы применяют для разметки Web-страницы и использование заливки и фоновых рисунков позволяет разработать интересный дизайн документа.

# Границы таблицы

По умолчанию обозреватели не показывают границы таблицы. Если же границы нужны, то их ширину можно установить с помощью специальных атрибутов, присвоив им значение в пикселях:

**BORDER** – внешняя граница таблицы;

**CELLPADDING** – горизонтальная граница;

**CELLSPACING** – вертикальная граница.

**BORDER='0'** – границы нет.

По умолчанию цвет границ – серый. Чтобы изменить цвет границы, можно воспользоваться атрибутом **BORDERCOLOR**, который изменит цвет как внешних, так и внутренних границ.

# Выборочная прорисовка границ

□ **FRAME** – внешние границы таблицы:

- **void** – нет внешних границ;
- **above** – граница по верхнему краю;
- **below** – граница по нижнему краю;
- **rhs** – граница по правому краю;
- **lhs** – граница по левому краю;
- **hsides** – границы по верхнему и нижнему краям;
- **vsides** – границы по правому и левому краям;
- **bolder** – все внешние границы.

□ **RULES** – внутренние границы таблицы:

- **none** – нет внутренних границ;
- **cols** – границы между столбцами;
- **rows** – границы между строками;
- **all** – все внутренние границы.

# Объединение ячеек

```
<td width='100' height='300' valign='left'
colspan='2'>&nbsp;</td>
<!--объединены 2 столбца-->
```

```
<td rowspan='2' width='100' height='300'
valign='left'>&nbsp;</td>
<!--объединены 2 строки таблицы-->
```

Для объединения ячеек в дескрипторах **<TD>** и **<TH>** устанавливают следующие параметры:

**COLSPAN** – число столбцов, занятых ячейкой;

**ROWSPAN** – число строк, занятых ячейкой.

Следует помнить, что при объединении ячеек число столбцов или строк сокращается на 1.



# Группирование столбцов и строк

Для группировки столбцов применяются парные дескрипторы:

`<COLGROUP>` - разбивка на столбцы разных типов (например, столбец заголовков и столбцы данных;

`<COL>` - произвольная группировка.

В обоих дескрипторах протяженность групп задается атрибутом `SPAN`. Этот атрибут объединяет указанное количество первых столбцов. Примененный еще раз, он объединяет следующие столбцы:

`<COLGROUP SPAN='2'> </COLGROUP> <!--объединены 1 и 2 столбцы>`

`<COLGROUP SPAN='2'> </COLGROUP> <!--объединены 3 и 4 столбцы>`

# Приложения

[Словарь дескрипторов  
\(тегов\)](#)

Сценарии на языке  
JavaScript

Готовые конструкции

[Начало страницы](#)

[Таблица](#)

[Таблица 1](#)

[Таблица 2](#)

[Таблица 3](#)

[Фрейм 1](#)

Фрейм 2

Фрейм 3

Таблицы стилей

[Глобальная](#)

[Связанная](#)

На содержание



# Литература

Рева О.Н. HTML. Просто как дважды два. – М.: Изд-во Эксмо, 2006  
Алленова Наталья – <http://www.postroika.ru>