

# **ПРОЕКТИРОВАНИЕ ВЕБ-СТРАНИЦ И ИНТЕРНЕТ-ГАЗЕТ**

**◆ ИЗДАТЕЛЬСТВО ТГТУ ◆**

Министерство образования и науки Российской Федерации  
Государственное образовательное учреждение  
высшего профессионального образования  
"Тамбовский государственный технический университет"

**ПРОЕКТИРОВАНИЕ  
ВЕБ-СТРАНИЦ И  
ИНТЕРНЕТ-ГАЗЕТ**

Методические указания  
для студентов 4 курса специальности 030602



---

Тамбов  
Издательство ТГТУ  
2005

УДК 004.738.52  
ББК ←973.202я73-5  
М38

Утверждено Редакционно-издательским советом ТГТУ

Р е ц е н з е н т

Кандидат технических наук, доцент  
*А.В. Лагутин*

С о с т а в и т е л и:

*С.А. Машков, С.Г. Машкова, И.А. Елизаров*

МЗ  
8 Проектирование Веб-страниц и Интернет-газет: Метод. указ. / Сост.: С.А. Машков, С.Г. Машкова, И.А. Елизаров. Тамбов: Изд-во Тамб. гос. тех. ун-та, 2005. 24 с.

Представлены основные сведения по разработке Интернет-страниц с помощью языка HTML. Приведено подробное описание тегов с примерами.

Предназначены для студентов 4 курса дневного отделения специальности 030602.

УДК 004.738.52  
ББК ←973.202я73-5

© Тамбовский государственный  
технический университет  
(ТГТУ), 2005

Учебное издание

ПРОЕКТИРОВАНИЕ ВЕБ-СТРАНИЦ И  
ИНТЕРНЕТ-ГАЗЕТ

Методические указания

С о с т а в и т е л и:

МАШКОВ Сергей Александрович,  
МАШКОВА Светлана Геннадьевна,  
ЕЛИЗАРОВ Игорь Александрович

Редактор З.Г. Чернова

Инженер по компьютерному макетированию М.Н. Рыжкова

Подписано к печати 28.10.2005.

Формат 60 × 84/16. Бумага газетная. Печать офсетная.

Гарнитура Times New Roman. Объем: 1,4 усл. печ. л.; 1,4 уч.-изд. л.

Тираж 100 экз. С. 693

Издательско-полиграфический центр  
Тамбовского государственного технического университета  
392000, Тамбов, Советская, 106, к. 14

## Введение

В течение последних лет российский Интернет из узкоспециализированной компьютерной среды превратился в мощное информационное пространство, охватывающее в той или иной степени все аспекты общественной жизни России.

Для специалиста по связям с общественностью Интернет предоставляет новые возможности и перспективы. Интернет стал крупнейшим мировым источником информации, в сети появились электронные версии традиционных газет и собственно сетевые издания.

Изучив предлагаемый материал, Вы сможете создавать собственные Веб-страницы, а затем и целые Интернет-газеты, оформлять их с помощью таблиц и фреймов, включая нестандартные компоненты, добавляя звук, графику и т.д.

## Основы HTML

Чтобы опубликовать в *Интернете документ* (под документом понимается файл, содержащий некоторую информацию и подготовленный с использованием специального *языка*), достаточно поместить его на компьютер-сервер, постоянно подключенный к *Интернету* и способный общаться с другими серверами с помощью *протокола передачи гипертекста* (HyperText Transfer Protocol, или http://).

Совокупность таких серверов получила название "всемирной паутины" (WWW или World Wide Web). WWW – это система организации информации в Интернете, которая позволяет объединять в одном структурированном документе (Веб-странице) информационные элементы различного происхождения (текст, изображения, звук), а также включать в любой документ ссылки на другие документы, расположенные в произвольных местах сети (гиперссылки). Набор Веб-страниц, связанных между собой ссылками и предназначенных для достижения единой цели, называется Веб-сайтом.

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. *Браузер* – это программа обмена информацией между компьютером и Веб-сервером и отображения ее на экране. Подготовленный (размеченный) документ нужно поместить на Веб-сервер, способный общаться с другими серверами по протоколу http.

HTML не является языком программирования, он только управляет расположением информации в окне браузера.

Основными понятиями HTML являются: тег, элемент, атрибут.

*Тег* – это символьная конструкция из открывающей и закрывающей угловых скобок – < >, между которыми находится конкретный символ или строка символов, предписывающие браузеру отображение последующего содержания документа в соответствии с их назначением.

Язык HTML использует различные вводимые в текстовые документы теги, которые указывают, каким образом информация должна считываться Веб-браузером. Большинство тегов – парные. Размещаемая информация помещается между открывающим тегом и закрывающим. Отличие закрывающего тега – наличие прямой косой черты после открывающей угловой скобки, например, </P >. Символы внутри угловых скобок тегов допускается записывать прописными или строчными буквами, так как браузеры не чувствительны к регистру, в котором набраны названия тегов.

*Элемент* – это контейнер из открывающего и закрывающего тегов, внутри угловых скобок которых находится название элемента в виде символа или строки символов, а между тегами – фрагмент документа, подлежащий интерпретации и отображению браузером в соответствии с назначением элемента. Существуют элементы, состоящие только из одного открывающего тега. Например, тег перевода строки <BR>.

*Атрибут* – это ключевое слово внутри открывающего тега элемента, имеющее стандартное имя и определенный набор значений, придающих те или иные свойства фрагменту документа, находящемуся между открывающим и закрывающим тегами элемента. В открывающем теге элемента может быть несколько атрибутов. Они отделяются друг от друга пробелами.

## Обязательные теги HTML

- <HTML> и </HTML> – определяет начало и конец *HTML*-документа.
- < HEAD > и </HEAD> – определяют начало и конец заголовка документа. В заголовок документа обычно включается наименование документа и множество дополнительной служебной информации.
- <TITLE> и </TITLE> – теги для определения наименования документа. Текст, помещенный между ними, воспринимается браузером как название документа и отображается им в заголовке окна, а при выводе на принтер – в левом верхнем углу каждой страницы.
- <BODY> и </BODY> – теги для определения тела *HTML*-документа. Тело документа отвечает и за информационное содержание и за внешний вид информации, представленной в окне браузера. Тег < BODY > может содержать несколько необязательных атрибутов:
  - BGCOLOR – определяет цвет фона. Цвет может быть именованным (“RED”, “BLUE”, “GREEN” и др.) или определяться в виде кода RGB (“#FF0000”, “#00FF50”). Например, <BODY BGCOLOR=“RED”>
  - BACKGROUND – позволяет задать фон *HTML*-страницы в виде графического изображения (рисунка). Если рисунок меньше окна браузера, то он многократно дублируется, заполняя все окно. Например, <BODY BACKGROUND = “рисunok.jpg”>, где рисunok.jpg – имя файла рисунка.
  - TEXT – определяет цвет текста *HTML*-страницы.
  - LINK – цвет гипертекстовой связи (ссылки).
  - VLINK – цвет уже посещенной ссылки.
  - ALINK – цвет активной ссылки. Активной является ссылка, к которой подводится указатель мыши.

*Простейший пример HTML-страницы:*

```
<HTML>
<HEAD>
<TITLE>Моя первая страница</TITLE>
</HEAD>
<BODY>
Простейший пример HTML-страницы, содержащий обязательные теги
</BODY>
</HTML>
```

*Пример простейшей цветной HTML-страницы:*

```
<HTML>
<HEAD>
<TITLE>Моя первая страница</TITLE>
</HEAD>
<BODY BGCOLOR=“#505050” TEXT=“YELLOW”>
Простейший пример HTML-страницы, содержащий обязательные теги с отображением желтого текста
на темно синем фоне
</BODY>
</HTML>
```

## Управление структурой документа

*HTML* позволяет разбивать текст на абзацы, начинать текст с новой строки, организовывать заголовки текста различного уровня, отделять различные фрагменты текста горизонтальной линией различной толщины, ширины и цвета. Для этих целей служат следующие теги:

- <H1>...</H1> – <H6>...</H6> – теги для организации заголовков шести уровней. Заголовок первого уровня самый крупный.
- <P> и </P> – теги для описания абзацев. Текст, помещенный между парой этих тегов, воспринимается как абзац и отделяется от других фрагментов текста дополнительной пустой строкой. Красная строка отсутствует.

Теги <P> и <H1> (i=1...6) могут содержать необязательный атрибут ALIGN, задающий выравнивание текста. Возможные значения этого атрибута: LEFT (выравнивание по левому краю), CENTER (по центру), RIGHT (по правому краю). Например:

<H1 ALIGN="CENTER"> Заголовок </H1>.

Центрирование любых элементов HTML-документа может быть осуществлено также с помощью тега <CENTER>, </CENTER>.

➤ <DIV> и </DIV>, <SPAN> и </SPAN> – теги для указания специальных свойств отдельному фрагменту текста (текстовому блоку).

Между тегами <DIV> и <SPAN> имеются существенные отличия. <DIV> является исключительно структурным тегом. Он создает принудительный перенос строки на одну позицию после своего закрывающего тега, поэтому задавать с его помощью отдельные свойства фрагмента внутри абзаца нельзя – это вызовет принудительный перенос строки. А <SPAN> берет начало в области физического форматирования текста, он позволяет назначать новые правила отображения текстовых фрагментов без изменения структуры документа.

➤ <BR> – этот непарный тег используется, если необходимо перейти на новую строку, не прерывая абзаца.

➤ <HR> – позволяет вывести горизонтальную разделительную линию.

Возможные атрибуты тега <HR>:

- SIZE – определяет толщину линии в пикселах (SIZE=4).
- WIDTH – задает размах линии по ширине экрана. Ширина может быть задана в абсолютных единицах – пикселах (WIDTH=400) или относительных – в процентах от ширины экрана (WIDTH=75%).
- COLOR – цвет разделительной линии.

## Форматирование шрифта

Существует два основных подхода к шрифтовому выделению фрагментов: логическое и физическое форматирование.

Основные теги физического форматирования шрифта представлены в табл. 1.

Теги физического форматирования указывают браузеру как нужно отображать тот или иной фрагмент текста (подчеркнутым, наклонным и т.д.) независимо от его смысловой нагрузки. Если необходимо показать, что фрагмент текста является, например, кодом программы или цитатой, можно использовать теги логического форматирования текста. Основные теги логического форматирования представлены в табл. 2.

### 1 Теги физического форматирования

Теги	Описание
<B>...</B>	Полужирный текст
<I>...</I>	Курсив
<U>...</U>	Подчеркнутый текст
<TT>...</TT>	Телетайпный текст
<STRIKE>...</STRIKE>	Перечеркнутый текст
<SUB>...</SUB>	Нижний индекс
<SUP>...</SUP>	Верхний индекс
<BIG>...</BIG>	Шрифт, увеличенный относительно стандартного размера
<SMALL>...</SMALL>	Шрифт, уменьшенный относительно стандартного размера

## 2 Теги логического форматирования

Теги	Описание
<CODE>...</CODE>	Пример исходного текста программы. Отображается телетайпным шрифтом
<SAMP>...</SAMP>	Используется для демонстрации образцов сообщений, выводимых на экран программами. Отображается телетайпным шрифтом
<KBD>...</KBD>	Используется для указания того, что нужно ввести с клавиатуры. Отображается телетайпным шрифтом
<VAR>...</VAR>	Используется для написания имен переменных. Отображается курсивом
<ADDRESS>, </ADDRESS>	Используется для указания того, что текст представляет собой почтовый адрес. Отображается курсивом
<STRONG>...</STRONG>	Выделенный текст. Отображается полужирным шрифтом
<EM>...</EM>	Выделенный текст. Отображается курсивом
<CITE>...</CITE>	Цитата. Отображается курсивом

Для форматирования шрифта также может использоваться парный тег <FONT>...</FONT>, который позволяет настроить начертание, размер и цвет шрифта. Этот тег имеет следующие атрибуты:

- SIZE= – задает размер шрифта (от 1 до 7, 1 – самый мелкий, 7 – самый крупный);
- COLOR= – определяет цвет шрифта;
- FACE= – задает начертание шрифта (“Arial”, “Courier”, “Tahoma” и т.д.).

### Предварительно отформатированный текст и текст с отступом

Тег-контейнер предварительно отформатированного текста <PRE>, </PRE> позволяет выводить на экран монитора текст такого формата, каким он представлен в окне текстового редактора (со всеми переносами строк, отступами, абзацами и прочим форматированием). Такая возможность весьма удобна, например, при отображении фрагментов программного кода "как есть", без его интерпретации браузером.

Визуально результат действия тега <PRE> выглядит как текст, набранный моноширинным шрифтом.

Парный тег <BLOCKQUOTE>, </BLOCKQUOTE> позволяет оформить текст, заключенный в нем, с отступом от начала строки. Этот тег удобно использовать для длинных цитат (в отличие от элемента CITE), при этом цитируемый текст отображается отдельным абзацем с увеличенным отступом. *Например:*

Редакция газеты благодарит Иванова Ивана Ивановича за его присланное очередное поздравление:

<BLOCKQUOTE>

С прекрасным праздником весны<BR>

Я вас поздравляю.<BR>

Добра, сердечной теплоты <BR>

И счастья вам желаю.<BR>

</BLOCKQUOTE>

### Отступы и специальные символы





Атрибут TYPE, помещенный в теге <UL> определяет тип маркера для всех элементов списка. Также возможно определение конкретного маркера для определенного элемента списка, в этом случае атрибут TYPE должен находиться в теге <LI>.

*Пример маркированного списка:*

```
<UL TYPE="CIRCLE">
  <LI> апельсины
  <LI> яблоки
  <LI TYPE="SQUARE"> груши
</UL>
```

В этом примере маркеры первых двух элементов списка будут отображаться в виде незакрашенных окружностей, а последний элемент – в виде закрашенного квадрата.

*Пример нумерованного списка:*

```
<OL>
  <LI> элемент списка – 1
  <LI> элемент списка – 2
</OL>
```

Нумерация в списке может быть осуществлена арабскими цифрами, большими и малыми римскими цифрами, заглавными и строчными латинскими буквами. За конкретный вид нумерации отвечает атрибут TYPE. Возможные значения атрибута TYPE:

- 1 – обычные арабские числа 1, 2, 3 и т.д. (значение по умолчанию);
- I – римские большие цифры I, II, III, IV и т.д.;
- i – римские малые цифры i, ii, iii, iv и т.д.;
- A – латинские заглавные буквы A, B, C и т.д.;
- a – латинские строчные буквы a, b, c и т.д.

Можно начать нумерацию с любой цифры или буквы. За это отвечает атрибут START тега <OL>. Например, при использовании тега

```
<OL TYPE="1" START="5">
```

нумерация списка будет начинаться с числа 5 (5, 6, 7 и т.д.), а при теге

```
<OL TYPE="A" START="5">
```

нумерация будет начинаться с буквы "E", так как "E" – пятая буква латинского алфавита.

Можно изменять естественный порядок нумерации в списке. За это отвечает атрибут VALUE тега <LI>. Например, необходимо начать нумерацию в списке с числа 5, а, начиная с третьего элемента продолжить нумерацию с числа 10. Этот список будет выглядеть следующим образом:

```
<OL START="5">
  <LI> элемент списка – 1
  <LI> элемент списка – 2
  <LI VALUE="10"> элемент списка – 3
  <LI> элемент списка – 4
</OL>
```

Списки могут быть вложенными, т.е. элемент списка может содержать другой список. Например, необходимо создать такой список

1. Фрукты:

- Апельсины

- Мандарины
2. Овощи:
- Помидоры
3. Другое

Этот список является вложенным и реализуется посредством размещения тегов нумерованных списков внутри тегов нумерованного.

```
<OL>
  <LI> Фрукты:
    <UL>
      <LI> Апельсины
      <LI> Мандарины
    </UL>
  <LI> Овощи:
    <UL>
      <LI> Помидоры
    </UL>
  <LI> Другое
</OL>
```

В HTML имеется третий вид списков – *список определений*, с их помощью удобно представлять различного рода словари, глоссарии. Организуются они с помощью парного тега `<DL>...</DL>`, тегов `<DT>` и `<DD>`.

- `<DL>...</DL>` – определяют начало и конец списка определений или терминов;
- `<DT>` – определяемый термин;
- `<DD>` – определение термина.

*Пример списка определений:*

```
<DL>
  <DT> Термин 1
  <DD> Определение термина 1
  <DT> Термин 2
  <DD> Определение термина 2
</DL>
```

Для создания маркированных фрагментов текста кроме списков могут быть использованы меню и каталоги. Для этого используются парные теги:

- `<MENU>...</MENU>` – создание меню.
- `<DIR>...</DIR>` – создание каталога.

Каждый новый элемент меню или каталога определяется тегом `<LI>`.

## Гиперссылки

В отличие от обыкновенного текста, который можно читать от начала к концу, гипертекст позволяет осуществлять переход от одного фрагмента текста к другому. В HTML переход от одного текста к другому задается с помощью парного тега `<A>...</A>`, который содержит обязательный атрибут `HREF`. Переход может быть осуществлен к другому документу, к конкретному месту текущего или другого документа (переход к анкеру (закладке)). При этом документ, на который осуществляется ссылка, может быть открыт в текущем или другом окне.

- `<A HREF="адрес перехода">` выделенный фрагмент текста `</A>` – создание ссылки.

- `<A HREF="адрес перехода#закладка">` выделенный фрагмент текста`</A>` – создание ссылки на закладку в другом документе.
- `<A HREF="#закладка">` выделенный фрагмент текста`</A>` – создание ссылки на закладку в том же документе.
- `<A HREF="адрес перехода" TARGET="***">` фрагмент текста`</A>` – создание ссылки на другое окно.

Для определения закладки (анкера) в том или ином документе используются тег `<A>...</A>` с атрибутом NAME:

- `<A NAME = "закладка">...</A>` – определить закладку.

*Пример.* Необходимо из некоторого файла (file1.html) осуществить переход к файлу file2.html к закладке, содержащей, например, текст “Прогноз погоды”. Оба файла находятся в одной директории.

В файле file2.html должна быть организована в требуемом месте (содержащем текст “Прогноз погоды”) закладка, с именем label1:

```
<A NAME=" например label1"> Прогноз погоды </A>
```

В файле file1.html может быть описано следующим образом:

```
<A HREF="file2.html#label1">узнать прогноз погоды</A>
```

Если переход осуществляется к документу, находящемуся на другом сервере, необходимо указать полный путь до этого документа, например:

```
<A HREF="http://www.tstu.ru/win/main.htm">ТГТУ</A>
```

В HTML-документе, кроме ссылок на Веб-документы, возможны ссылки на другие виды ресурсов Интернета. Например, ссылка к информации, расположенной на ftp-сервере, может иметь вид:

```
<A HREF="ftp://des.tstu.ru/pub/doc/">Загрузить</A>
```

Также можно использовать электронную почту:

```
<A HREF="mailto: ipu@ahp.tstu.ru">послать письмо</A>.
```

Если пользователь совершит переход по такой ссылке, у него на экране откроется окно ввода исходящего сообщения его почтовой программы. В строке To: (Куда:) окна почтовой программы будет указан адрес назначения ipu@ahp.tstu.ru.

### **Использование графики**

Графические изображения вставляются в HTML документ с помощью одиночного тега IMG, содержащий обязательный атрибут SRC, значением которого является адрес графического файла, например:

```
<IMG SRC="рисunok.jpg">
```

При такой записи файл рисунка должен находиться в той же директории что и HTML документ. Если файл рисунка находится в другой директории, то должен быть указан абсолютный или относительный адрес этого файла. Абсолютный адрес включает имя сервера и путь до этого файла на сервере, например:

```
<IMG SRC="www.tstu.ru/win/pict/risunok.jpg">
```

Относительный адрес представляет адрес, относительно текущего HTML документа. Если текущий HTML документ находится в директории win, то с использованием относительной адресации, рисунок может быть вставлен в HTML документ помощью тега

```
<IMG SRC="/pict/risunok.jpg">
```

Тег <IMG> также может содержать следующие необязательные атрибуты:

- ALT= "альтернативный текст"> – используется для вывода поясняющего рисунок текста. Отображается этот текст в месте вставки рисунка и при наведении на рисунок указателя мыши.
- WIDTH= – ширина изображения в пикселах.
- HEIGHT= – высота изображения в пикселах.
- HSPACE= – горизонтальный отступ от графического изображения.
- VSPACE= – вертикальный отступ от графического изображения.
- BORDER= – толщина рамки вокруг изображения в пикселах.
- ALIGN= – выравнивание изображения. Возможные значения атрибута:
  - LEFT, RIGHT – выравнивание по левому и правому полю;
  - MIDDLE или CENTER – выравнивание по центру;
  - TOP, BOTTOM – выравнивание по верху и низу текущей строки.
- TITLE= "подсказка"– атрибут, определяющий своим текстовым значением подсказку, которая будет появляться при задержке указателя мыши над ссылкой.

Рисунок можно сделать кнопкой, при нажатии на которую будет осуществлен переход к другому HTML документу. Это осуществляется, например, конструкцией вида:

```
<A HREF="file2.html"><IMG SRC="risunok.jpg"></A>
```

## Таблицы

Таблицы имеют большое значение при создании Веб-страниц. Таблицы могут использоваться для создания правильной структуры сайта и отображения информации (невидимые таблицы).

Таблица определяется тегами

➤ <TABLE> и </TABLE>, внутри которых находятся все элементы и данные таблицы.

Тег <TABLE> имеет следующие необязательные атрибуты:

- BORDER = – задает толщину окантовки таблицы. При BORDER=0 (значение по умолчанию) таблица становится невидимой.
- CELSPACING= – устанавливает расстояние между ячейками в пикселах.
- CELLPADDING= – устанавливает для всех ячеек отступ данных от рамки ячейки. Значение указывается в пикселах.

- WIDTH = – определяет ширину всей таблицы. Ширина может быть указана в пикселах (WIDTH=“300”) или в процентах от ширины экрана (WIDTH=“75%”).
- HEIGHT = – определяет высоту всей таблицы.
- ALIGN = – устанавливает выравнивание таблицы:
  - LEFT – выравнивание таблицы по левому краю;
  - RIGHT – выравнивание таблицы по правому краю;
  - CENTER выравнивание таблицы по центру.
- BGCOLOR = – определяет цвет фона всей таблицы.
- BACKGROUND = – задает адрес фонового рисунка таблицы.
- BORDERCOLOR = – определяет цвет рамки таблицы.

Каждая таблица состоит из строк и столбцов, или иначе – ячеек, расположенных в строке.

➤ <TR>и </TR> – определяют строку таблицы. Количество строк всей таблицы определяется количеством встречающихся пар тегов <TR> и </TR>.

➤ <TD>и </TD> – определяют ячейку таблицы. В каждую строку, образуемую тегами <TR> и </TR>, вкладывается столько тегов <TD> и </TD>, сколько ячеек необходимо получить в данной строке.

➤ <TH> и </TH> – определяют заголовочную ячейку таблицы. Отличие от тегов <TD> и </TD> состоит в том, что в ячейке формируемой тегами <TH> и </TH> текст будет отображаться по центру полужирным шрифтом.

Теги <TR>, <TD> и <TH> аналогично тегу <TABLE> могут содержать атрибуты WIDTH, HEIGHT, BGCOLOR, BACKGROUND, ALIGN. Атрибут ALIGN отвечает за горизонтальное выравнивание данных ячеек и позволяет выровнять по левому, правому краю или по центру данные для всех ячеек строки или для конкретной ячейки.

Также теги <TR>, <TD> и <TH> могут содержать атрибут VALIGN, определяет вертикальное выравнивание содержимого всех ячеек строки или конкретной ячейки таблицы. Возможные значения атрибута VALIGN:

- TOP – выравнивание по верхней границе ячейки (строки);
- BOTTOM – выравнивание по нижней границе ячейки (строки);
- MIDDLE – выравнивание по центру ячейки (строки).

Таблица может иметь название, которое задается тегами <CAPTION> и </CAPTION >. Тег <CAPTION> имеет атрибут ALIGN, отвечающий за положение названия таблицы:

- ALIGN = “TOP” – название расположено сверху таблицы;
- ALIGN = “BOTTOM” – название расположено снизу таблицы.

*Пример.* Необходимо реализовать таблицу (см. рисунок) с размерами ячеек по горизонтали 100 и 200 пикселей.

Таблица

Ячейка 1	Ячейка 2
Ячейка 3	Ячейка 4

```
<TABLE ALIGN= “center” border=1 width=300>
<CAPTION>Таблица</CAPTION>
<TR>
  <TD width=100>Ячейка 1</TD>
  <TH width=200>2</TH>
</TR>
<TR>
  <TD width=100>Ячейка 3</TD>
  <TD ALIGN=“center” width=200>Ячейка 4</TD>
```

```
</TR>
</TABLE>
```

Язык HTML позволяет создавать гораздо более сложные таблицы, в которых имеется горизонтальное и вертикальное объединение ячеек. Осуществляется это с помощью атрибутов COLSPAN и ROWSPAN тегов <TD> или <TH>:

- COLSPAN = – устанавливает размах ячейки по горизонтали, т.е. количество ячеек, которые будут объединены по горизонтали, начиная с указанной ячейки.
- ROWSPAN = – устанавливает размах ячейки по вертикали, т.е. количество ячеек, которые будут объединены по вертикали вниз, начиная с указанной ячейки.

*Пример.* Необходимо реализовать следующую таблицу:

1	2		
	3	4	
5	6	7	8

```
<TABLE ALIGN= "center" border=1>
<TR>
  <TD ROWSPAN = "2">1</TD><TD COLSPAN = "3">2</TD>
</TR>
<TR>
  <TD>3</TD><TD COLSPAN = "2">4</TD>
</TR>
<TR>
  <TD>5</TD><TD>6</TD><TD>7</TD><TD>8</TD>
</TR>
</TABLE>
```

## Фреймы

Фреймы – средство для разделения экрана на несколько областей, в каждой из которых отображается содержимое отдельной Веб-странички. Каждый фрейм имеет свой URL, что позволяет загружать его независимо от других фреймов. Каждый фрейм имеет собственное имя, позволяющее переходить к нему из другого фрейма. Размер фрейма может быть изменен пользователем прямо на экране при помощи мыши (если это специально не запрещено). Использование фреймов в большинстве случаев позволяет существенно улучшить внешний вид Веб-страницы.

Фрейм-документ является специфичным видом HTML-документа – он не содержит тега <BODY>. Этот документ описывает только фреймы, их размеры, положение, загружаемые в них HTML-документы.

Создание фреймов осуществляется с помощью тегов:

➤ <FRAMESET> и </FRAMESET>.

Этот тег имеет следующие атрибуты:

- COLS= – подразделяет экран на определенное количество вертикальных колонок [в кавычках через запятую указывается желаемый размер (ширина) каждой колонки]. Размер может быть задан в абсолютных единицах, в процентах от размера экрана (%) или отмечен символом звездочка (\*). Звездочка означает, что все оставшееся место в окне будет отдано под соответствующую колонку. Например, записи “20%, 30%, 50%” и “20%, \*, 50%” полностью аналогичны.
- ROWS= – делит экран на определенное количество горизонтальных колонок.

- BORDERCOLOR= – определяет цвет границы между фреймами.
- BORDER= – задает ширину границы между фреймами в пикселах.
- FRAMEBORDER= – определяет наличие видимой границы между фреймами. Возможные значения этого атрибута “YES” и “NO”.
- FRAMESPACING= – позволяет задать расстояние между фреймами.

После создания фреймов их необходимо определить с помощью тега <FRAME> внутри тегов <FRAMESET> и </FRAMESET>.

Тег <FRAME> имеет следующие атрибуты:

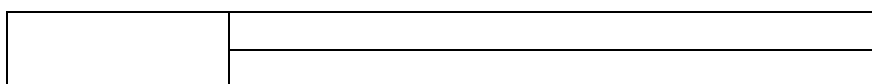
- SRC = – определяет адрес отображаемой во фрейме Веб-страницы. Этот атрибут является обязательным.
- NAME = – имя данного фрейма.
- SCROLLING = – атрибут, управляющий полосой прокрутки: “YES” – полосы прокрутки присутствуют всегда, “NO” – отсутствует, “AUTO” – появляется в случае необходимости.
- MARGINWIDTH= и MARGINHEIGHT= – устанавливают расстояние (в пикселах) между границей фрейма и его содержимым соответственно по горизонтали и вертикали.
- NORESIZE – отключает возможность изменения размера фрейма.

*Пример фрейм-документа (файл index.html):*

```
<HTML>
<HEAD><TITLE>фрейм-документ</TITLE>
</HEAD>
<FRAMESET ROWS="20%,*,50%" BORDER="5">
  <FRAME SRC="file1.html" NAME="frame1" NORESIZE>
  <FRAME SRC="file2.html" NAME="frame2">
  <FRAME SRC="file3.html" NAME="frame3">
</FRAMESET>
</HTML>
```

В этом примере организовано три вертикальных фрейма соответственно с размерами 20, 30 и 50 % от экрана с шириной границы 5 пикселей. Первый фрейм не допускает изменения его размеров с помощью мыши. В каждый из фреймов будут загружены соответственно файлы file1.html, file2.html, file3.html.

Фреймы могут быть вложенными. Это обеспечивается вложением тегов <FRAMESET> и </FRAMESET>. Необходимо, например, экран задать следующим образом:



```
<HTML>
<HEAD><TITLE>Вложенные фреймы</TITLE></HEAD>
<FRAMESET ROWS="20%,80%">
  <FRAME SRC="file1.html" NAME="frame1">
  <FRAMESET COLS="50%,50%">
    <FRAME SRC="file2.html" NAME="frame2">
    <FRAME SRC="file3.html" NAME="frame3">
  </FRAMESET>
</FRAMESET>
</HTML>
```

### Мета-определения

Мета-определения электронного документа описываются целым рядом параметров, входящих в состав непарного тега <META> и предназначенных для описания внутренних свойств HTML-файла.



Все мета-определения, в сущности, имеют два основных типа данных: HTTP-EQUIV и NAME. Структура мета-тега с использованием HTTP-EQUIV выглядит следующим образом:

<META HTTP-EQUIV="имя" CONTENT="содержание">

Наиболее часто используемые кодовые конструкции данного мета-определения представлены в табл. 3.

### 3 Конструкции мета-определения META HTTP-EQUIV

Значение HTTP-EQUIV	Действие	Подпараметр CONTENT
Refresh	Перезагрузка/переадресация через заданный промежуток времени	URL
Content-Type	Определение типа и кодировки документа	charset
Content-Language	Указание языка документа	
Location	Указание места расположения документа в Интернете (полный адрес)	

Конструкция META HTTP-EQUIV="refresh" указывает браузеру перезагружать содержимое окна через заданный промежуток времени. При добавлении дополнительного параметра URL через указанное время (в секундах) будет произведена переадресация на внешний адрес. Следующий пример дает браузеру команду ровно через 10 секунд перенаправить посетителя по адресу <http://www.tstu.ru>:

<META HTTP-EQUIV="refresh" CONTENT="10; URL=http://www.tstu.ru">

Причин для использования такой конструкции может быть несколько: ваш Веб-сайт переехал на другой сервер, вы хотите показать рекламный блок, а потом перенаправить посетителя на главную страницу и т.д.

Конструкция META HTTP-EQUIV="Content-Type" позволяет задать тип и кодировку документа. Наиболее используемыми значениями кодировки русскоязычных HTML-документов являются Windows-1251 и KOI8-R.

*Например:*

<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=Windows-1251">

Мета-теги с атрибутом NAME используются поисковыми роботами при индексации Вашего сайта. Структура мета-тега с использованием NAME выглядит следующим образом:

<META NAME="имя" CONTENT="содержание">

Наиболее часто используемые кодовые конструкции данного мета-определения представлены в табл. 4.

### 4 Конструкции мета-определения META NAME

Значение NAME	Действие
Key-	Задаёт набор ключевых слов документа, предна-

words	значенных для индексирования поисковыми системами. Слова указываются через запятую
Description	Предоставление небольшого описания текущего HTML-документа, также необходимого для поисковых систем
Author	Предназначен для указания автора текущего документа
Robots	Используется при описании правил для индексирования документа поисковыми системами (роботами)
URL	Указание основного адреса документа для индексирования (в случае существования "зеркальных" копий)
copyright	Установление авторского права и перечисление условий распространения документа
generator	Информация о программном обеспечении

*Примеры использования мета-тегов с атрибутом NAME:*

```
<META NAME="keywords" CONTENT="HTML, Веб, WWW, Веб-страница, Интернет">
<META NAME="description" CONTENT="HTML. Экспресс-курс: Быстрое освоение языка HTML">
<META NAME="Author" CONTENT="Иванов И.И.">
<META NAME="url" CONTENT="http://site.ru">
<META NAME="copyright" CONTENT="©2003-2004 TambovGSM">
<META NAME="generator" CONTENT="Macromedia FireWorks">
```

## Карты-изображения

На многих HTML-документах сегодня с успехом используются так называемые карты-изображения (Imagemaps), которые представляют собой обычные графические файлы (как правило, стандарта GIF или JPEG) с привязанными к различным областям этого изображения гиперссылками. Такие области описываются специальными координатами, в соответствии с которыми браузер переносит пользователя на нужную страницу.

Для определения конфигурации активных областей карты-изображения используется специальный тег-контейнер <MAP> с параметром NAME, который должен соответствовать свойству параметра USEMAP в теге <IMG>. Необходимо отметить, что описание карты-изображения следует сразу же после указания тега рисунка <IMG>:

```
<IMG SRC="picture.gif" USEMAP="#mymap">
<MAP NAME="mymap">
```

Координаты активных областей...

```
</MAP>
```

Описание активных областей карты-изображения осуществляется с помощью тега <AREA>, не требующего за собой закрывающего тега.

Тег <AREA> может включать следующие параметры:

- shape – определяет форму активной области (значения такие же, как в формате CERN серверного варианта);
- coords – осуществляет выбор конкретной активной области и содержит значения пар координат. Начало координат размещается в верхнем левом углу графического изображения, которому соответствует значение 0,0;
- target – используется при работе с фреймами;

- alt – действие параметра аналогично использованию его в обычных гиперссылках на основе графических указателей (альтернативный текст);
- href – наличие гиперссылки для данной области;
- nohref – отсутствие гиперссылки для данной области.

*Пример карты-изображения:*

```
<IMG SRC="box.jpg" BORDER="1" WIDTH="300" HEIGHT="234" ALT="gift" USEMAP="#gift">
<MAP NAME="gift">
<AREA SHAPE="rect" COORDS="60, 26, 222, 98" HREF="bantik.html" ALT="Бантик">
<AREA SHAPE="rect" COORDS="63,88,135,188" HREF="left.html" ALT="Левая сторона">
<AREA SHAPE="rect" COORDS="151,102,225,200" HREF="right.html" ALT="Правая сторона">
</MAP>
</BODY></HTML>
```

## Встраивание звуковых файлов в HTML-документ

Для встраивания практически любых звуковых файлов в HTML-документ можно использовать два пути:

- создание ссылки на музыкальный файл и размещение ее на странице;
  - применение специального тега вставки музыкальных файлов <embed>.
- В HTML-документы встраиваются, как правило, следующие форматы звуковых файлов:
- WAV – формат звуковых файлов операционной системы Microsoft Windows;
  - MIDI – Musical Instrument Digital Interface (формат электронных инструментов);
  - AU – формат Sun/NeXT;
  - AIFF – формат платформы Macintosh;
  - RealAudio – формат передачи звуковых файлов в режиме реального времени.

### *Создание гиперссылки на музыкальный файл*

Правила записи гиперссылки на файлы музыкальных форматов абсолютно идентичны записи ссылок на HTML-документы или графику:

```
<A HREF="music.wav">Музыкальный файл (30 Кб)</A>
```

При нажатии на такую ссылку на компьютере пользователя запускается установленный по умолчанию проигрыватель звуковых файлов.

### *Применение тега <EMBED>*

Тег <embed> имеет ряд параметров, которые позволяют управлять воспроизведением музыкального файла непосредственно в окне браузера.

*Пример встраивания звукового файла с помощью тега <embed>:*

```
<EMBED SRC="melody.wav" WIDTH="287" HEIGHT="213" AUTOSTART="true">
```

В этом случае браузер отобразит встроенную панель воспроизведения указанного музыкального файла ("melody.wav").

Параметры тега <embed> приведены в табл. 5.

### 5 Параметры тега <embed>

Параметр	Описание
SRC	Указание пути к воспроизводимому звуковому файлу форматов AU, AIFF, WAV и MIDI.
width	Ширина панели воспроизведения (обязательно)

height	Высота панели воспроизведения (обязательный параметр)
Autostart	Значение по умолчанию – FALSE. Для автоматического начала воспроизведения используется значение true
autoload	Значение по умолчанию – true. Для запрета автоматической загрузки файла используется значение false
align	Тип выравнивания панели воспроизведения относительно текста на странице. Возможные значения: LEFT, RIGHT, CENTER, TOP и baseline.
volume	Установка громкости воспроизведения
starttime	Время начала воспроизведения в минутах и секундах (формат записи: "mm: ss"; по умолчанию – 00.00)
endtime	Время окончания воспроизведения в минутах и секундах (формат записи: "mm: ss "; по умолчанию – конец звукового файла)
controls	Указание элементов управления на панели воспроизведения (значения console, smallconsole, playbutton, pausebutton, stopbutton, volumelever). Значение по умолчанию – console
console	Отображение полного набора элементов управления на панели воспроизведения

### **Фоновый звук**

В завершение разговора о музыкальных форматах, следует упомянуть о такой возможности, как вставка фонового звука в Веб-документ:

```
<BODY BGSOUND="audio/intro.wav" LOOP="infinite">
```

Данный пример задает для страницы фоновый звук "intro.wav" с помощью параметра bgsound. Конструкция LOOP="infinite" дает браузеру команду проигрывать указанный файл бесконечное количество раз. Для ограниченного количества воспроизведения фонового звука в качестве значения необходимо использовать целые числа (1, 2, 3 и т.д.).

### *Лабораторная работа 1*

### **Первое знакомство с HTML**

*Цель работы:* Ознакомиться с основными понятиями HTML, разработать простейшие HTML страницы.

*Задание:* Ознакомиться со структурой HTML документа. Изучить обязательные теги <HTML>, <HEAD>, <TITLE>, <BODY>. Изучить теги управления разметкой документа: <P>, <H1>-<H6>, <BR>, <HR>. Изучить теги форматирования шрифтов. Создать в редакторе Notepad несколько простейших Веб-страниц, содержащих перечисленные теги.

## **Списки**

*Цель работы:* Изучить принципы построения списков.

*Задание:* Изучить теги для создания маркированных, нумерованных списков и списков определения; создать списки с различными видами маркеров и видами нумерации. С помощью необходимых атрибутов изменить порядок нумерации в списке. Создать вложенные списки, сочетая при этом различные виды списков.

## *Лабораторная работа 3*

### **Гиперссылки**

*Цель работы:* Изучить принципы построения гиперссылок.

*Задание:* Изучить тег <A> и его атрибуты для создания гиперссылок на другие HTML-документы. Научиться производить переход на конкретные элементы Веб-страниц (переход по анкеру).

## *Лабораторная работа 4*

### **Использование графики в Веб-страницах**

*Цель работы:* Изучить принципы построения иллюстрированных Веб-страниц.

*Задание:* Изучить тег <IMG> и его атрибуты для вставки графики в HTML-документы. Научиться использовать рисунки в качестве управляющих кнопок гиперссылок.

## *Лабораторная работа 5*

### **Таблицы**

*Цель работы:* Получить навыки использования таблиц в Веб-страницах.

*Задание:* Изучить теги <TABLE>, <TR>, <TD>, <TH> и их атрибуты. Создать простые таблицы и таблицу с наличием горизонтального и вертикального объединения ячеек (конкретный вид таблиц задается преподавателем). Использовать невидимые таблицы для улучшения структуры представляемой информации.

## *Лабораторная работа 6*

### **Фреймы**

*Цель работы:* Получить навыки использования фреймов при создании Веб-страниц.

*Задание:* Изучить теги <FRAMESET>, <FRAME> и их атрибуты. Создать многооконные Веб-страницы с использованием простых и вложенных фреймов. Создать статические фреймы и поместить в них управляющие кнопки. С помощью атрибута TARGET тега <A> научиться осуществлять переходы между фреймами.

## *Лабораторная работа 7*

### **Мета-определения**

*Цель работы:* Получить навыки использования Мета-определений в Веб-страницах.

*Задание:* Изучить Мета-определения, научиться пользоваться основными данными Мета-определений – HTTP-EQUIV и NAME. Научиться задавать тип и кодировку документа, языка документа. Задать в HTML-документе Мета-теги с атрибутом NAME для индексации Вашей страницы в Интернете.

## *Лабораторная работа 8*

### **Использование карт-изображений**

*Цель работы:* Изучить принципы построения карт-изображений в Веб-страницах.

*Задание:* Изучить создание карт-изображений при помощи серверного и клиентского варианта. Научиться создавать карты-изображения при помощи клиентского варианта с привязанными к различным областям этого изображения гиперссылками.

## **Лабораторная работа 9**

### **Встраивание звуковых файлов в HTML-документ**

*Цель работы:* Изучить принципы встраивания звуковых файлов в HTML-документ.

*Задание:* Создать гиперссылки на музыкальный файл. Изучить тег для вставки музыкальных файлов `<embed>` и его атрибуты. Научиться использовать тег `<embed>` для вставки музыкальных файлов и для управления ими. Изучить вставку фонового звука в HTML-документ при помощи тега `<BODY>` и атрибута `BGSOUND`.

### **Список рекомендуемой литературы**

- 1 Кириленко А. Самоучитель HTML. СПб.: Питер, 2005.
- 2 Комолова Н.В. HTML: Учебный курс. Изд. 1-е. СПб.: Питер, 2005.
- 3 Моррисон М. Книга HTML и XML. Быстро и эффективно. СПб.: "Питер", 2005.
- 4 Петюшкин А. Профессиональное программирование HTML в Web-дизайне. СПб.: "БХВ-Петербург", 2004.
- 5 Пол МакФедрис. Язык HTML. М.: Юнити, 1996.