



федерация
интернет
образования

С. Ю. Севастьянов

ВВЕДЕНИЕ В ЯЗЫК HTML

◆ ИЗДАТЕЛЬСТВО ТГТУ ◆

Министерство образования Российской Федерации
Тамбовский региональный центр Федерации Интернет-Образования

С. Ю. Севастьянов

ВВЕДЕНИЕ В ЯЗЫК HTML

Учебно-методическое пособие

Тамбов
◆ Издательство ТГТУ ◆
2002

УДК 800.92(075)
ББК 3973-018.2я73
С28

Севастьянов С. Ю.
С28 Введение в язык HTML: Учебно-метод. пособ. Тамбов: Изд-во
Тамб. гос. техн. ун-та, 2002. 32 с.

Учебно-методическое пособие "Введение в язык HTML" предназначено для слушателей курсов повышения квалификации на базе Тамбовского РЦ ФИО по программе "Интернет-технологии для учителя предметника", "Интернет-технологии для администратора образовательного учреждения".

УДК 800.92(075)
ББК з973-018.2я73

© Тамбовский региональный центр

© Севастьянов С. Ю., 2002

© Тамбовский государственный
технический университет (ТГТУ), 2002

Федерации Интернет-Образования, 2002

Учебное издание

СЕВАСТЬЯНОВ Сергей Юрьевич

ВВЕДЕНИЕ В ЯЗЫК HTML

Учебно-методическое пособие

Редактор З. Г. Чернова

Инженер по компьютерному макетированию М. Н. Рыжкова

ЛР № 020851 от 27.09.99

П_рп № 020079 от 28.04.97

Подписано в печать 05.04.2002.

Гарнитура Times New Roman. Формат 60×84/16.

Бумага офсетная. Печать офсетная. Объем: 1,86 усл. печ. л.; 1,7 уч.-изд. л.

Тираж 100 экз. С. 257^М.

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

Введение	5
.....	
1 Язык HTML и его реализация	7
.....	
1.1 Что такое HTML	7
1.2 Программы просмотра HTML-страниц	8
2 Основы языка	10
.....	
2.1 Структура HTML-страницы	10
2.2 Простейшие элементы языка	12
2.3 Ссылки	15
2.4 Подробный пример	16
3 О возможных ошибках	18
.....	
4 ВАЖНЕЙШИЕ КОНСТРУКЦИИ ЯЗЫКА	20
4.1 <BODY> – тело документа	20
4.2 , <I>, <BIG> и – управление шрифтами ...	21
4.3 <HR> – горизонтальная черта	23
4.4 <PRE> – предварительно форматированный текст	24
4.5 <SUB> и <SUP> – индексы	24
4.6 <U> – подчеркивание	24
4.7 <TITLE> – заголовок	25
4.8 Таблицы	25
5 ручная подготовка документов и Редакторы HTML	29
.....	

Данное "Введение в язык HTML" первоначально было написано как электронный учебник. Сделанное на этом же языке, оно было разбито на отдельные страницы, имело многочисленные ссылки между ними.

Данный "бумажный" вариант соответствует электронной версии 2.2.

К сожалению, полное соответствие обеспечить не удалось – возможности интерактивной работы намного богаче, чем простое чтение. Тем не менее, основная часть содержимого сохранена в неизменном виде.

*Отзывы и пожелания можно направлять автору
или по телефону (0752) 72–07–35.*

по электронной почте seva@tambov.ru

ВВЕДЕНИЕ

Работа в Internet у многих ассоциируется с WWW-серверами, суть которых в предельно краткой форме можно описать следующим образом.

На экране отображена некоторая информация – текст, картинки, иногда – с элементами мультипликации. Может присутствовать и звуковое сопровождение. Отдельные места экрана выделены, например, голубым цветом и подчеркиванием. Если в такие выделенные места подвести курсор (он при этом меняет форму, превращаясь, скажем, в указующий перст) и нажать кнопку мыши, на экране появится новая информация. Она обычно также содержит выделения, ссылающиеся на новые порции информации, и т.д. В результате простым нажатием кнопки мыши удастся путешествовать по всему миру.

Для тех, кто в Internet не работал, не собирается, а если и работает, то не имеет ни малейшего намерения делать собственный сервер (и даже не знает, что это такое), рассмотрим пример, не имеющий отношения к сетям.

Нужно подготовить доклад, иллюстрируемый текстами, графиками, рисунками и т.п. на экране компьютера. Программировать Вы не умеете или не имеете времени на это. Ну что же, выходом является язык HTML.

Пусть, например, иллюстративный материал доклада имеет такой вид:



Страница 1 – начало. При последовательном изложении с нее надо перейти на страницу 2, далее – на страницу 3, и, наконец, на страницу 4. Однако может потребоваться с первой страницы сразу попасть на третью или четвертую. И хорошо, если бы было возможно возвращаться обратно на предыдущую страницу, перепрыгивать со второй страницы сразу на четвертую и т.д.

Язык HTML дает все эти возможности. И огромное количество других.

Ну а если доклад делать не надо? Нужна презентация материала, электронный учебник, книга, справочник... Да мало ли что еще можно сделать на HTML с минимальными затратами труда и времени.

Язык HTML дает огромные преимущества перед многими другими способами представления информации с помощью компьютера.

1 ЯЗЫК HTML И ЕГО РЕАЛИЗАЦИЯ

1.1 Что такое HTML

Термин HTML (HyperText Markup Language) переводится на русский как "Язык манипулирования гипертекстами".

Ключевым здесь является слово "гипертекст". Под ним подразумевается, в отличие от простого текста, возможность создания полноценных документов, содержащих разнообразные шрифты, графику, звук, что достигается путем включения в текст специальных указаний (их иногда называют "тэгами"). Структура тэгов четко оговорена в описании языка.

Выдача гипертекстового документа (или, как часто говорят, "страницы") на экран или бумагу производится в соответствии с этими указаниями. Однако в подавляющем большинстве случаев автор документа не оговаривает многие детали, считая, что они не критичны. Тогда программа, осуществляющая выдачу документа, использует запомненные ею – при создании или в процессе настройки пользователем – значения. Таким образом, одна и та же HTML-страница может отображаться по-разному на разных машинах и даже на одной машине в разные моменты времени.

Приведем примеры.

Обычно в HTML-странице не оговаривается шрифт текстовой части документа – не указано ни название шрифта, ни его размер. Ясно, что выдача на машине с установленным по умолчанию значением размера шрифта в 10 пунктов будет выглядеть совершенно не так, как на машине со шрифтом в 18 пунктов.

Даже при одинаковых шрифтах в окнах разного размера страница будет отличаться – станет длиннее в узком окне (попробуйте сейчас и посмотрите).

Конечно, можно указать полностью все параметры страницы. Но, во-первых, зачем? Сильно ли изменится впечатление от страницы на экране монитора, если шрифт будет выглядеть немного не так? Или Вы указали очень мелкий шрифт, а у того, кто смотрит вашу страницу, похуже монитор (или зрение...).

Во-вторых, нет гарантии, что все ваши указания будут выполнены буквально. Скажем, уверены ли Вы, что используемый Вами экзотический шрифт установлен на всех машинах, где будут читать документ? Или что все экраны позволят иметь заданную Вами ширину страницы?

Так что оставим побольше свободы читающему – ведь HTML-страницы в основном предназначены не для бумаги, а для экрана.

Второй важнейшей особенностью языка HTML является возможность включения в документ ссылок на другие документы. Внешне ссылка выглядит как выделенное (например, цветом или подчеркиванием) место текста или рисунка. Поставив указатель мыши на это место и нажав кнопку, мы получим на экране новый документ – тот, который был указан в ссылке. Таким образом, появляется возможность переходить с одного документа на другой, возможно, даже расположенный на другой машине (разумеется, если ваша машина подключена к сети).

1.2 Программы просмотра HTML-страниц

Возможно, этот раздел Вам неинтересен или непонятен – тогда прочитайте первый абзац и смело [переходите дальше](#).

Чтобы выдать на экран или напечатать HTML-страницу, нужна специальная программа – браузер. Поскольку язык HTML весьма распространен, написано довольно много таких программ, чем-то (иногда сильно) отличающихся между собой. Большинство из них ориентированы на работу в оконной графической среде.

Я умышленно написал это по-русски, а не употребил общепринятый английский термин Windows, так как в России, к сожалению, он у большинства работающих с компьютерами ассоциируется с цифрами 95/98/2000... Windows'ов существует довольно много, и 95/98 не относится к лучшим из них. Есть X-Windows, OpenWin, MacOS, есть в конце концов Windows 3.1 той же фирмы Microsoft, несравнимо с Windows-95 лучший по надежности.

Рассмотрим кратко самые ходовые браузеры.

Netscape Navigator фирмы Netscape. Самый распространенный до недавнего времени в мире браузер. Тот, кто попробовал с ним поработать, вряд ли откажется от него в пользу других. Может работать практически в любой среде – Windows 3.1, начиная с оригинала, Windows95/98, Windows NT, OpenWin, X-Windows, Macintosh. Для отечественного пользователя имеет очень важное свойство – распространяется бесплатно (совершенно официально, с разрешения фирмы!).

Internet Explorer фирмы Microsoft. В последние годы ощутимо потеснил позиции Navigator'a, главным образом благодаря агрессивной рекламе фирмы. Поскольку делает его Microsoft, то и работает он только в среде MS Windows – 95/98, 3.1, NT. Бесплатный – как сыр в мышеловке – при условии, что Вы уже оплатили Windows. Достоинство (а скорее, причина широкого распространения) – входит в базовый комплект поставки MS Windows.

NCSA Mosaic фирмы NCSA. Самый первый из браузеров, и самый компактный (я запускал его на 286-й машине с 1 мегабайтом памяти). Долгое время был абсолютно бесплатным, позднее появились коммерческие версии. К сожалению, больше не развивается и, как следствие, не поддерживает некоторые из самых свежих достижений языка HTML, поэтому в последнее время используется мало.

Lynx. Полностью бесплатный продукт, без поддержки графики и звука. В текстовом режиме экрана, без всяких окон, позволяет смотреть гипертекст и, главное, ссылки в нем. Работает под Unix, есть версия под DOS. Распространяется в виде исходного текста. Применимость ограничена случаем, когда нужно "побегать" по ссылкам, а ресурсы ограничены (например, на терминале нет графики).

Каждый браузер имеет отличия от других, приводящие в некоторых случаях к отличиям в отображении HTML-документов. Небольшая разница в изображении в общем-то непринципиальна, но, к сожалению, иногда разработчики вносят собственные дополнения в язык HTML, в результате чего документы, сделанные под конкретный браузер, невозможно увидеть в других. Особенно грешит этим фирма Microsoft, поэтому, даже используя их Explorer, лучше не пользоваться расширениями языка. Кстати, их расширения, сделанные для языка Java, также поддерживаемого Explorer'ом, привели к тому, что распространение вирусов по сети существенно упростилось – для машин, на которых работает Explorer.

2 ОСНОВЫ ЯЗЫКА

В этой главе приводятся начальные сведения по языку HTML. Их достаточно для создания простейших страниц и структуры ссылок между ними.

В принципе, можно этим и ограничиться. Однако практика показывает, что почти каждый, успешно сделав несколько страниц, стремится к улучшению их оформления.

В настоящее время доступно большое количество литературы, в том числе на русском языке. Можно поискать такие руководства, а также официальное описание языка в Internet.

2.1 Структура HTML-страницы

Большинство элементов языка HTML по своей сути являются скобками. Открывающая скобка отмечает начало части страницы, обладающей некоторыми свойствами, закрывающая – конец этой части. Все, что находится между скобками, одинаково с точки зрения этих свойств. Например, можно задать для части текста цвет, отличающийся от цвета остальных частей.

Скобками элементы являются именно по сути, а по форме это не скобки в традиционном виде, а конструкции следующего вида:

```
<слово>  
выделяемая скобками часть  
</слово>
```

<слово> задает начало выделяемой части и вид конкретных скобок (например, для изменения цвета можно задать), а то же слово в тех же знаках < >, но с дробной чертой перед словом – окончание этой части (). Список возможных слов фиксирован в описании языка.

Вся HTML-страница заключается в пару скобок

```
<HTML>  
</HTML>
```

(Еще раз подчеркнем, что каждая скобка – это не один знак, а все, что находится между символами < >, и сами эти символы.)

Итак, все, что находится между скобками <HTML> и </HTML> – это HTML-страница. Каждая страница должна храниться в отдельном файле.

Страница состоит из двух частей – заголовка и тела.

Заголовок задает оформление страницы в целом, и может вообще отсутствовать. Так что отложим его рассмотрение на следующие главы, а здесь просто отметим, что начало заголовка отмечается скобкой <HEAD>, а конец – скобкой </HEAD>.

А вот тело отсутствовать не должно – что же это получится за страница, на которой ничего нет.

Начинается тело страницы скобкой <BODY>, заканчивается – </BODY>.

Теперь мы можем записать общую структуру страницы в таком виде:

<HTML>
<HEAD>
заголовок
</HEAD>
<BODY>
тело страницы
</BODY>
</HTML>

Если Вы поняли этот раздел, то уже сейчас можете создать простейшую HTML-страницу, в которой будет текст и ничего более. Создайте файл с любым именем и расширением. HTML, любым текстовым редактором (например, встроенным в Norton Commander или редактором Записная книжка из стандартного комплекта MS Windows 3.x/9x/2000), поместите в него приведенные элементы структуры, вместо слов "тело страницы" напечатайте свой текст – и страница готова. Смотрите ее своим браузером (Netscape, Internet Explorer, Mosaic и т.д.). В качестве заготовки можно воспользоваться файлом s1.htm в оглавлении SAMPLES данного руководства.

А теперь перейдем к [рассмотрению элементов тела страницы](#).

2.2 Простейшие элементы языка

2.2.1 Абзацы

Если Вы попытались создать простейший вариант страницы, как это было описано в предыдущем разделе, то уже заметили, что границы строк, напечатанных Вами, полностью игнорируются браузером. В частности, не осталось и следа от разбиения на абзацы.

Так и должно быть. Все браузеры воспринимают текст файла как непрерывный поток слов, полностью игнорируя границы строк, величину интервала между словами, отступы для абзацев и заголовков и т.д. Все эти действия задаются специальными элементами языка, а расположены они вместе с текстом в файле могут быть произвольным образом – хоть в единственной суперстроке, хоть по одному слову на каждой строчке. Возможны и все промежуточные варианты, главный критерий – как удобно Вам, лишь бы слова были отделены одно от другого, по крайней мере, одним пробелом или переходом на новую строку.

Абзац задается включением в текст явного указания на него. Имеются несколько вариантов, например:

**
**
<P>

Первый из них задает просто переход на новую строку, второй указывает на новый абзац. К сожалению, большинство браузеров оформляют абзацы в форме, почти не используемой в русскоязычной литературе – разделение абзацев без отступа, но с пропуском пустой строки.

Поскольку переход на новую строку выполняется в конкретном месте, задавать закрывающие скобки **</BR>** и **</P>** нет необходимости. Впрочем, если они заданы, скажем, в конце абзаца, то никак помешать не могут.

2.2.2 Заголовки

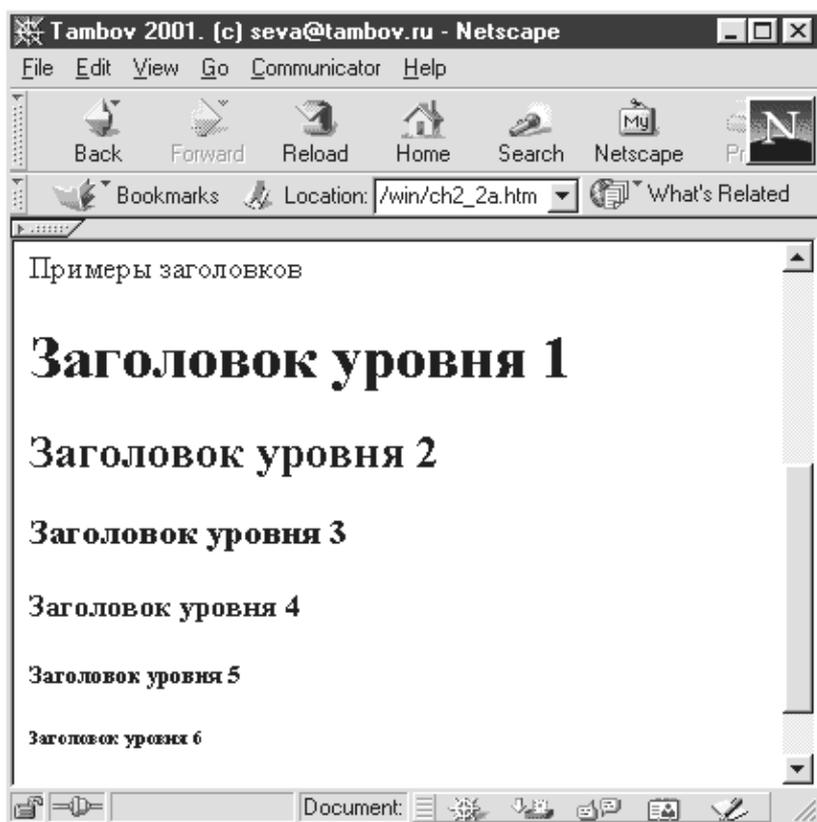
Как и в печатном тексте, заголовки могут отличаться размером букв и другими характеристиками в зависимости от степени важности.

В HTML определены 6 уровней заголовка, нумеруемых по убыванию важности от 1 до 6 и обозначаемых скобками:

<H1> </H1>
<H2> </H2>
<H3> </H3>
<H4> </H4>
<H5> </H5>
<H6> </H6>

Весь текст внутри скобок считается заголовком и выдается соответственно настройке вашего браузера.

Вот как это выглядит в моем Netscape 4.75:



Для некоторых браузеров выделение заголовка непосредственно не означает перехода на новую строку, так что если он требуется (как чаще всего это и бывает), его лучше указать явно с помощью `<P>` или `
`.

2.2.3 Центрирование

В приведенном выше примере все заголовки были прижаты к левому краю окна. Если же заголовок (или какой-либо иной элемент, скажем, картинку) надо расположить посередине строки, применяются скобки

`<CENTER> </CENTER>`

То, что заключено между ними, будет отцентрировано.

Например, конструкция

`<H3><CENTER>Это пример</CENTER></H3>`

с вложенными скобками дает следующий результат:

Это пример

2.2.4 Вставка иллюстраций

В HTML-страницу можно вставить графический файл, хранимый отдельно. Способ подготовки файла не критичен – можно отсканировать изображение, нарисовать его любым графическим редактором, сохранить образ экрана монитора специальной программой-граббером и т.д. Допустимые форматы файлов (от них зависит расширение в имени) определяются браузером. Наиболее рекомендуемы форматы JPG (для полутоновых изображений) и GIF (для штриховых), поддерживаемые всеми браузерами. Форматами BMP и PCX лучше не пользоваться – браузеры их не любят (видны только в Explorer'e).

Для вставки картинки в конкретное место страницы используется указание

``

Здесь мы впервые сталкиваемся с элементом языка, в котором ключевое слово после знака `<` задает только вид элемента, а детализация указана другими ключевыми словами, отделенными от первого и последующих пробелами. (Здесь только два слова, но их бывает и больше).

Картинка вставляется именно там, где она указана, и обычно смещена влево. Но ничто не мешает ее отцентрировать. Не забывайте только указать переход на новую строку сразу за ней.

Рассмотренных элементов языка достаточно для создания полноценных страниц, поэтому описание других элементов оформления отложим, а займемся способами [перехода с одной страницы на другую](#).

2.3 Ссылки

Простейший переход – возврат на предыдущую страницу – не требует никаких указаний на самой странице, а выполняется средствами браузера. Переход же на любую страницу по желанию автора возможен, если в явном виде задать, куда надо перейти. Для этого служит конструкция

```
<A HREF="имя файла">  
отображаемая информация  
</A>
```

Скобки <A HREF> – задают начало и конец ссылки. После ключевого слова HREF указано имя файла, который будет отображен в случае нажатия мышью... куда? На текст, заключенный между скобками. Именно этот текст выделяется цветом и подчеркиванием в браузере.

Между скобками можно поставить не только текст, но и картинку. Тогда внутри скобок <A HREF> – появится указание на картинку . Подчеркивание и выделение цветом здесь вряд ли будет смотреться хорошо, поэтому браузер обычно ограничивается голубой рамкой вокруг рисунка. Совсем без выделения плохо – трудно понять, что это ссылка.

2.4 Подробный пример

Для иллюстрации всего изложенного в этом разделе вернемся к приведенному во [введении](#) примеру.



Назовем для определенности 4 файла, содержащих 4 страницы, именами

p1.htm
p2.htm
p3.htm
p4.htm

(О расширениях для HTML-файлов раньше речь не заходила. Лучше всего иметь расширение .html, но поскольку в MS DOS длина расширения ограничена тремя символами, можно обойтись значением .htm, тем более что браузеры не возражают против этого.)

Далее приведены полные тексты этих страниц.

Файл p1.htm:

```
<HTML>  
<BODY>  
<p>Страница 1  
<P>Текст страницы...  
<P><A HREF="p2.htm">Ссылка на страницу 2</A>  
<P><A HREF="p3.htm">Ссылка на страницу 3</a>  
<P><A HREF="p4.htm">Ссылка на страницу 4</A>  
</BODY>  
</HTML>
```

Файл p2.htm:

```
<HTML>  
<BODY>  
<P>Страница 2
```

```
<P>Текст страницы...
<P><A HREF="p3.htm">Ссылка на страницу 3</A>
<P><A HREF="p1.htm">Вернуться на страницу 1</A>
</BODY>
</HTML>
```

Файл p3.htm:

```
<HTML>
<BODY>
<P>Страница 3
<P>Текст страницы...
<P><A HREF="p4.htm">Ссылка на страницу 4</a>
<P><A HREF="p1.htm">Вернуться на страницу 1</a>
</BODY>
</HTML>
```

Файл p4.htm:

```
<HTML>
<BODY>
<P>Страница 4
<P>Текст страницы...
<p><A HREF="p1.htm">Вернуться на страницу 1</A>
</BODY>
</HTML<
```

3 О ВОЗМОЖНЫХ ОШИБКАХ

Непонятные конструкции языка браузером просто игнорируются. Так сделано умышленно: ведь основное применение браузера – сеть Internet, а работающий в сети, просматривая HTML-документы на чужой машине, не должен получать от своего браузера многочисленные сообщения из-за чужих ошибок.

Так что практически все ошибки не могут привести ни к чему фатальному – страница будет видна. Но результат, скорее всего, будет существенно отличаться от ожидаемого.

При изготовлении собственных страниц Вы наверняка будете ошибаться, и анализ причин ошибки может стать довольно непростым делом. Поэтому попытаемся указать хотя бы некоторые типичные случаи.

Отсутствие закрывающей скобки приводит к тому, что выделенный элемент текста продолжается дальше, чем это предусмотрено автором. Например, текст, выделяемый как ссылка скобками `<a href> – `, при отсутствии `` остается выделенным до конца страницы или до следующей ссылки. В последнем случае срабатывает закрывающая скобка второй ссылки, но зато сама эта ссылка отсутствует.

Ссылка не срабатывает в случае неверного указания файла, на который мы ссылаемся. В этом случае браузер выдает сообщение об ошибке. Интересный эффект иногда получается при полном отсутствии имени или указании вместо имени знака вопроса или звездочки – выдается одно из оглавлений диска.

Вставляемый в документ графический файл может не отображаться и заменяться каким-нибудь условным значком вроде разорванной картинки по ссылке. А может быть, используемый Вами формат не поддерживается браузером. Выглядит это, например, так:



Если не отображаются все картинки на всех настройками браузера – ему просто запрещено отмечены места, где они должны быть). Что ж, придется менять настройки.

Довольно сложен для диагностики случай **непарных кавычек**, например, при указании имени файла. Внешние проявления могут быть самыми разнообразными, поскольку следующие открывающие кавычки воспринимаются закрывающими к первой паре, и далее вся структура документа полностью нарушается. При этом выпадают (то есть не отображаются) большие куски документа. Нужно искать самое начало таких

страницах, это скорее всего связано с выдавать картинки (обычно просто

искажений, а затем внимательно смотреть исходный текст.

Русские буквы отображаются закорючками в случае указания отсутствующего на машине шрифта либо при несовпадении кодировки для редактора и браузера. Для тех, кто не знает: о кодировке русских букв можно написать целую книгу и даже не одну. В сетях шли такие споры, что от мордобоя участников удерживали только разделяющие их тысячи километров. Если же говорить коротко, имеется около десятка базовых способов хранения русских букв (кириллицы) в компьютере и их неисчислимые вариации. Часто даже в одной машине используются разные способы одновременно! Например, если Вы напечатали свой HTML-документ с использованием встроенного в Norton Commander редактора, в браузере под Windows Вы вряд ли увидите кириллицу. Тем не менее, я рекомендую пользоваться этим редактором, а затем обрабатывать файл прилагаемой к Руководству программой. Так что давайте и дальше чтить Кирилла и Мефодия...

Закорючки не только вместо кириллицы, а вообще вместо всего, по крайней мере их много в начале документа. Вероятно, Вы готовили HTML-документ редактором Word. Не делайте этого! Вам нужен чистый текст, а Word пишет в файл огромное количество собственной служебной информации. Чтобы спасти положение, коль файл уже готов, можно попробовать превратить его в HTML встроенными в Word средствами, а потом поправить в нормальном текстовом редакторе. О редакторах, делающих чистый текст без всяких вставок [будет дальше](#).

4 ВАЖНЕЙШИЕ КОНСТРУКЦИИ ЯЗЫКА

Предлагаемый в этой главе набор конструкций в достаточной мере произволен. Автор не ставил целью написать **полное** руководство по языку – речь идет именно о введении.

4.1 <BODY> – тело документа

Приведенный ранее простейший вариант элемента <BODY>, состоящий только из ключевого слова, позволял только указать начало тела документа. Для задания различных способов украшения страницы в целом используется ряд параметров элемента BODY, задаваемых в виде пар

ключевое-слово=значение

указываемых после слова BODY, но перед знаком >, как это было рассмотрено для и <A> ранее.

Рассмотрим некоторые из них.

Цвет фона страницы задается словом

BGCOLOR=цвет

В качестве цвета можно указать либо английское название одного из 16 цветов, либо шестнадцатичное значение любого из более чем 16 миллионов практически произвольных цветов.

В языке HTML фиксированными названиями цветов являются:

BLACK	Черный	OLIVE	Оливковый
TEAL		RED	Красный
BLUE	Синий	MAROON	Коричневый
NAVY	Темно-синий	GRAY	Серый
LIME	Светло-зеленый	FUCHSIA	
GREEN	Зеленый	PURPLE	Фиолетовый
SILVER	Серый	YELLOW	Желтый
AQUA	Голубой	WHITE	Белый

Следует учитывать, что некоторые браузеры обрабатывают гораздо больше названий цветов. Не следует на это надеяться, да и запомнить сотни названий не так-то просто.

Гораздо лучше при необходимости задания цвета, отсутствующего среди шестнадцати стандартных, задать его в виде тройки чисел в формате RGB – красный-зеленый-синий. Каждое из трех чисел задает интенсивность соответствующего цвета в относительных единицах от 0 до 255 (естественно, что большему значению соответствует большая интенсивность).

К сожалению, числа надо задавать в шестнадцатиричном виде, что удобно не для всех.

В этой форме цвет имеет вид #RRGGBB, где RR соответствует интенсивности красного, GG – зеленого и BB – синего.

Вместо цвета фона можно задать фоновую картинку в формате

BACKGROUND=имя-файла

Как и для иллюстраций, вставляемых в документ, желательно использовать только форматы GIF и JPG.

В случае одновременного указания параметров BGCOLOR и BACKGROUND картинка перекрывает фоновый цвет, который частично виден только при использовании прозрачного GIF.

Задание цвета текстовой части документа задается параметрами

TEXT (основной текст документа)

LINK (текст в ссылках)

VLINK (ссылки, просмотренные ранее)

ALINK (ссылки, на которые в данный момент производится указание, то есть этот цвет возникает в момент нажатия на клавишу мыши и исчезает после ее отпущения)

Способ задания цвета ничем не отличается от задания цвета фона.

4.2 , <I> <BIG> и – управление шрифтами

Любой фрагмент текста страницы можно **выделить жирным шрифтом**, заключив его в пару скобок

** – **

Для задания курсива используются скобки

<I> – </I>

Крупный шрифт задается с помощью

<BIG> – </BIG>

Возможно одновременное применение нескольких элементов. Например, ***жирный курсив*** получается вложением

<I> – </I>

При этом важно, чтобы внутренние скобки были закрыты раньше, чем внешние. При пересечении последствия получаются непредсказуемыми. Поэтому приведенный выше вариант является правильным, а вот такой:

<I> – </I> – неправильным.

Если рассмотренных возможностей по управлению шрифтами недостаточно, можно применить элемент , имеющий вид

выделяемый текст

Слово COLOR задает цвет букв, отменяя тем самым значение, указанное в BODY для выделенной части страницы. Способ задания цвета – как в BODY.

Слово SIZE указывает размер шрифта – абсолютный в виде числа без знака, либо приращение по отношению к базовому размеру (обычно выбираемому пользователем браузера) в виде числа со знаком. Естественно, что + задает увеличение, а – уменьшение шрифта. Относительный размер предпочтительнее, поскольку автор заранее не знает ни возможностей монитора у читающего страницу, ни его предпочтений.

Параметр FONT позволяет также задавать вид шрифта, однако автор категорически против этого по самой

тривиальной причине – никто еще не назвал ему хотя бы двух шрифтов, имеющихся на всех машинах.

4.3 <HR> – горизонтальная черта

Указание <HR> в любом месте текста вызывает переход на новую строку, причем вставляется горизонтальная линия.

Возможно задание толщины линии в пикселях в виде

<HR size=10>

Длина (ширина?) линии задается в процентах от ширины окна

<HR width=50%>

или в пикселях (тогда знак % просто не указывается).

Укороченную линию можно сдвинуть вправо, влево, либо отцентрировать, указав

<HR width=50% align=right>

<HR width=50% align=left>

<HR width=50% align=center>

Наконец, можно задать сплошную линию, указав

<HR size=5 noshade>

А вот задание цвета для HR стандартом не предусмотрено.

4.4 <PRE> – предварительно форматированный текст

В некоторых случаях в страницу требуется включить фрагмент текста, уже разбитого на строки. При этом из каких-либо соображений (часто просто от элементарной лени, автор знает это по себе...) вставка указаний языка HTML по форматированию нецелесообразна.

Достижение результата возможно указанием <PRE> непосредственно перед таким текстом и </PRE> после него.

Недостатком метода является расширение размера страницы сверх ширины окна в случае наличия в тексте хотя бы одной длинной строки, поэтому без крайней необходимости пользоваться предварительным форматированием не следует.

4.5 <SUB> и <SUP> – индексы

Задание в тексте нижних и верхних индексов выполняется заключением нужного фрагмента текста соответственно в скобки

_–

[–]

Результат:

Основной текст_{нижний индекс}
Основной текст^{верхний индекс}

4.6 <U> – подчеркивание

Для выделения части текста подчеркиванием используются скобки

<U> – </U>

Пользоваться подчеркиванием рекомендуется только тогда, когда это действительно необходимо, поскольку при чтении страницы подчеркнутый текст напоминает ссылку.

4.7 <TITLE> – заголовок

Это – единственный в данной главе элемент языка, относящийся не к телу документа, а к его заголовку.

Внутри скобок заголовка страницы <HEAD> – </HEAD> возможно указание наименования страницы, отображаемого в самой верхней строке окна. Оно заключается в пару скобок

<TITLE> – </TITLE>

4.8 Таблицы

Таблицы являются удобным и простым способом разделения частей страницы на горизонтальные и вертикальные фрагменты, выровненные между собой. Как и обычных таблицах, части могут быть отделены или не отделены друг от друга рамкой из вертикальных и горизонтальных линий.

Таблица целиком заключается в пару скобок

<TABLE>

</TABLE>

Указание закрывающей скобки для таблицы, как и для всех ее составных частей является очень важным. В случае незакрытой таблицы одни браузеры не отображают вообще ничего, другие искажают документ до неузнаваемости.

Клетки (ячейки) таблицы задаются в порядке горизонтальных рядов слева направо:

Ячейка 1	Ячейка 2	Ячейка 3
Ячейка 4	Ячейка 5	Ячейка 6

Каждый горизонтальный ряд заключается в скобки

<TR>

</TR>

а каждая ячейка внутри ряда в скобки

<TD>

</TD>

Таблица, приведенная выше, была задана так:

```
<table border=1>  
<tr><td>Ячейка 1</td>  
<td>Ячейка 2</td>  
<td>Ячейка 3</td></tr>  
<tr><td>Ячейка 4</td>  
<td>Ячейка 5</td>  
<td>Ячейка 6</td></tr>  
</table>
```

В этом примере для задания ширины линий рамки указан атрибут **BORDER**. Число после ключевого слова задает ширину рамки в пикселях. Значение **BORDER=0** указывает на отсутствие линий рамки.

В принципе изложенного достаточно для построения таблиц. Однако во многих случаях требуется оформить таблицу с учетом специфических требований автора, поэтому рассмотрим некоторые возможности 'украшения'

таблиц.

Для <TABLE> можно задать:

WIDTH=число или **WIDTH=процент%**

Здесь указывается желаемая ширина таблицы в пикселях или процентах от ширины окна, причем во втором случае целое число сопровождается знаком %.

Аналогично

HEIGHT=число или **HEIGHT=процент%**

задает желаемую высоту таблицы в пикселях или процентах.

Таблицу можно выровнять влево, вправо или по центру окна путем указания соответственно

ALIGN=LEFT

ALIGN=RIGHT

ALIGN=CENTER

Для <TD> можно указать свойства отдельной ячейки:

ALIGN=LEFT

ALIGN=RIGHT

ALIGN=CENTER

задает выравнивание по горизонтали текста внутри ячейки влево, вправо или по центру соответственно.

VALIGN=TOP

VALIGN=BOTTOM

VALIGN\MIDDLE

задает вертикальное выравнивание – по верхнему краю, по нижнему или посередине.

Оба эти атрибута могут быть указаны и в <TR>. В этом случае они определяют выравнивание для всех ячеек строки, кроме тех, где они же заданы явно.

Желаемая высота и ширина ячейки указываются с помощью параметров, аналогичных параметрам для таблицы:

WIDTH=число **WIDTH=процент%**

HEIGHT=число **HEIGHT=процент%**

Аналогично цвету фона и фоновому рисунку для страницы в целом можно задать их для ячейки таблицы. Параметры те же:

BGCOLOR=цвет

BACKGROUND=картинка

Рассмотрим в заключение главы два примера нетривиального применения таблиц.

1 Заголовок, окруженный картинками



Заголовок



Таблица из одной строки с тремя колонками, без рамки, шириной в 100 % окна. В первой ячейке выравнивание влево (картинка), в последней – вправо, во второй – по центру (текст).

```
<table border=0 width=100%>
```

```
<tr>
```

```
<td align=left><img src=cd.gif></td>
```

```
<td align=center><h1>Заголовок</h1></td>
```

```
<td align=right><img src=cd.gif></td>
</tr>
</table>
```

2 Кнопки с надписями

Назад
К началу
Вперед

Одна колонка, три строки. Рамки нет, сами ячейки окрашены, текст выделен жирным шрифтом и белым цветом.

```
<table border=0>
<tr align=center>
<td bgcolor=black><font color=white><b>Назад</b></font></td>
</tr>
<tr align=center>
<td bgcolor=black><font color=white><b>К началу</b></font></td>
</tr>
<tr align=center>
<td bgcolor=black><font color=white><b>Вперед</b></font></td>
</tr>
</table>
```

5 РУЧНАЯ ПОДГОТОВКА ДОКУМЕНТОВ И РЕДАКТОРЫ HTML

Часто приходится слышать, что можно делать гипертекстовые документы, совершенно не зная HTML. Что ж, некоторый резон в этом есть. Существует большое количество редакторов, позволяющих автору сосредоточиться на оформлении страницы прямо на экране, а рутинную работу по преобразованию в команды HTML берет на себя редактор.

Я не против. Но трудно добиться хороших результатов, не понимая сути процесса. Можно сделать прекрасную страницу, а для правильного указания ссылок уже нужно знать, что это такое. А может, ваша страница станет корявой при переходе с Explorer'a на Mosaic? Какими элементами языка пользовался редактор, Вы не знаете.

Кроме того, редакторы часто являются коммерческими продуктами. Бесплатное распространение еще не означает бесплатного использования. Проходит месяц, и установленный HotDog (один из таких редакторов) перестает работать. В этом смысле приятным исключением является редактор, встроенный в Netscape, начиная с версии Navigator 3 Gold.

Пользоваться редактором или нет, и каким конкретно – решать Вам. Но от использования в качестве такого редактора все того же MS Word рекомендую воздержаться. Мне приходилось видеть много людей, читающих HTML-страницы, сделанные в Word, броузерами не от фирмы Microsoft. Не знаю других обстоятельств, в которых матерятся так, как при этих попытках...

Никаких общих рекомендаций по пользованию другими редакторами здесь привести невозможно – каждый из них работает по-своему. Мой совет – попробуйте **Nescape Composer** (компонент Netscape Communicator), а потом подумайте, нужно ли Вам что-то еще.

Но если вы решили делать все вручную, то уделите внимание выбору текстового редактора.

Прежде всего, он действительно должен быть текстовым, то есть не добавлять к тексту никакой служебной информации. Иногда ее не видно, но тем не менее она есть, и может сильно помешать. Типичный пример – Lexicon; сколько я из-за него претерпел...

Чем можно пользоваться без всяких опасений?

1 Редакторы, встроенный в командеры – Norton Commander, Volkov Commander, Dos Navigator, Demos Commander, Far, MC и т.д. Вызываются они чаще всего простым нажатием клавиши F4 при курсоре, установленном на имя редактируемого файла. Все они очень просты, но как раз этим и привлекательны.

2 Norton Editor (вызывается командой NE). Он немного посложнее, но зато и дает больше возможностей, вроде перемещения частей файла или вставки в редактируемый файл другого.

3 Редактор "Блокнот" из базового комплекта MS Windows всех версий. Если Вы работаете в Windows, то он хорош уже тем, что из этой среды уходить не надо.

Список этот можно продолжить.

А чем пользоваться нельзя? Все перечислить просто невозможно (некоторые редакторы были упомянуты). Если Вы не знаете свойств редактора, можно дать только общий критерий: если в редакторе есть смена шрифтов, вставка картинок или другие возможности по украшению текста, почти наверняка он добавляет в файл что-то

