

* Язык

гипертекстовой

разметки HTML

- * ВВЕДЕНИЕ
- * 1. ОСНОВНЫЕ ПОНЯТИЯ HTML
- * 2. СТРУКТУРА HTML-ДОКУМЕНТА
 - * 2.1. Раздел документа HEAD
 - * Название документа TITLE
 - * 2.2. Раздел документа BODY
 - * 2.2.1. Спецификация элемента BODY
 - * 2.2.2. Советы по использованию атрибутов тега BODY
 - * 2.3. Примеры
 - * 2.3.1. Пример простого HTML-документа
 - * 2.3.2. Пример использования фонового рисунка
- * 3. ЦВЕТОВЫЕ СПЕЦИФИКАЦИИ
 - * 3.1. Формат RGB
 - * 3.2. Символьная нотация
 - * 3.3. Соответствие формата RGB и символьной нотации
- * 4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ
 - * 4.1. Абзацы
 - * 4.2. Управление переводом строки
 - * 4.3. Заголовки
 - * 4.4. Списки
 - * 4.4.1. Маркированный список
 - * 4.4.2. Нумерованный список
 - * 4.4.3. Список определений

* ОГЛАВЛЕНИЕ

- * 4.5. Форматирование текста
- * 4.5.1. Тег
- * 4.5.2. Контейнер DIV
- * 4.5.3. Отступы
- * 5. ГИПЕРССЫЛКИ
- * 5.1. Универсальный идентификатор ресурсов URL
- * 5.2. Правила записи ссылок
- * 5.3. Внутренние ссылки
- * 5.4. Ссылки на документы различных типов
- * 5.5. Ссылки на ресурсы Интернета
- * 6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ
- * 6.1. Горизонтальные линии
- * 6.2. Таблицы
- * 6.3. Рисунки
- * Рисунок-ссылка
- * 7. ЗАКЛЮЧЕНИЕ
- * ЛИТЕРАТУРА

ОГЛАВЛЕНИЕ

*

- * Internet входит в нашу жизнь. Единое информационное пространство оказывает все большее влияние на окружающий мир. Число пользователей Сети растет лавинообразно. Internet используют как малые, так и большие предприятия для рекламы, торговли, связи. Поэтому возрастает необходимость в специалистах, умеющих создавать и обслуживать Web-узлы. Цель настоящих методических указаний состоит в том, чтобы студенты получили основные навыки создания Web-страниц с помощью языка гипертекстовой разметки документов HTML.
- * Методические указания представляют собой описание основных элементов языка HTML. Для лучшего усвоения материала здесь приведены примеры и рекомендации по использованию элементов языка. Для закрепления изученного в конце каждой темы даны контрольные вопросы или задания.

*** ВВЕДЕНИЕ**

*

* **HTML** - язык гипертекстовой разметки документов (HyperText Markup Language). С помощью HTML создаются Web-страницы, которые находятся в глобальной компьютерной сети Интернет. HTML - это не язык программирования в традиционном смысле, он является языком разметки. С помощью HTML текстовый документ разбивают на блоки смысловой информации (заголовки, параграфы, таблицы, рисунки и т. п.).

* **Гипертекстовый документ** - это документ, содержащий переходы (**гиперссылки**) на другие документы. При использовании гиперссылки происходит перемещение от одного документа к другому (как по цепочке) в Интернете. HTML-документ является гипертекстовым документом.

* Особенности HTML-документа:

* 1. HTML-документ может содержать текст, графику, видео и звук.

* 2. В общем случае HTML-документ - это один или несколько текстовых файлов, имеющих расширение .htm или .html.

* 3. Создавать HTML-документ можно как с помощью специальных программ - редакторов HTML (например, FrontPage), так и с помощью любого текстового редактора (например, блокнота Windows).

* 4. Для просмотра HTML-документов существуют специальные программы-**броузеры**. Они **интерпретируют** HTML-документы, т.е. переводят текст документа в Web-страницу, и отображают ее на экране пользователя. Существует очень много различных броузеров, но наиболее распространенными броузерами являются Microsoft Internet Explorer, Netscape Navigator и Opera.

* 5. Если при интерпретации HTML-документа броузер чего-то не понимает, то сообщения об ошибке не возникает, а это место в HTML-документе игнорируется и не отображается броузером.

* HTML-документ состоит из элементов HTML.

* **Элемент HTML** - это чаще всего два тега (открывающий и закрывающий) и часть документа между ними. Кроме того, существуют элементы HTML, состоящие только из одного тега.

* **Тег** - в переводе с английского - ярлык, этикетка. Тег определяет тип выводимого элемента HTML (например, заголовок, таблица, рисунок и т.п.). Сам тег не отображается броузером. Тег представляет собой последовательность элементов:

* · символ левой угловой скобки (<) - начало тега;

* · необязательный символ слеша (/) - символ используется, чтобы обозначить закрывающий тег;

* · имя тега;

* · необязательные атрибуты в открывающем теге;

* · символ правой угловой скобки (>)

* **Атрибуты** - необязательный набор параметров, определяющих дополнительные свойства элемента HTML (например, цвет или размер).

Атрибут
состоит:

1. ОСНОВНЫЕ ПОНЯТИЯ HTML

* **Атрибуты** - необязательный набор параметров, определяющих дополнительные свойства элемента HTML (например, цвет или размер). Атрибут состоит:

- * · из имени атрибута;
- * · знака равенства (=);
- * · значения атрибута - строки символов, заключенной в кавычки

*

* Пример элемента HTML:

* `<H1 ALIGN= "CENTER">Глава 1</H1>`

*

* В этом примере:

* `<H1 ALIGN= "CENTER">` - открывающий тег

* `</H1>` - закрывающий тег

* H1 - имя тега

* ALIGN= "CENTER" - атрибут

* ALIGN - имя атрибута

* "CENTER" - значение атрибута

* Глава 1 - содержание элемента HTML

*

* Правила создания HTML-документов:

- * 1. Теги и атрибуты можно записывать в любом регистре, т.е. `</H1>` и `</h1>` - это одно и то же.
- * 2. Несколько пробелов подряд, символы табуляции и перевода строки при интерпретации браузером заменяются на один пробел. Это позволяет писать хорошо структурированные исходные тексты файлов HTML.
- * 3. Рекомендуется давать имена файлам HTML строчными английскими буквами. Длина имени - до восьми символов. В принципе

* 1. ОСНОВНЫЕ ПОНЯТИЯ HTML

- * Каждый HTML-документ должен начинаться тегом `<HTML>` и заканчиваться тегом `</HTML>`. Эти теги обозначают, что находящиеся между ними строки представляют единый HTML-документ. Кроме того, можно заметить, что файл HTML в целом является элементом языка HTML.
- * Также в HTML-документе должны присутствовать элементы HEAD (информация о документе) и BODY (тело документа).

*

* 2.1. РАЗДЕЛ ДОКУМЕНТА HEAD

*

- * Раздел документа HEAD определяет его заголовок, а также содержит дополнительную информацию о документе для браузера. Этот раздел не является обязательным, однако рекомендуется всегда использовать его в своих HTML-документах, так как правильно составленный заголовок может быть весьма полезен.
- * Раздел заголовка начинается тегом `<HEAD>` и следует сразу за тегом `<HTML>`. Между открывающим и закрывающим тегами элемента HEAD располагаются другие элементы заголовка.

*

* *Название документа TITLE*

*

- * Для того чтобы дать название HTML-документу, предназначен тег `<TITLE>`. Это название будет выведено в заголовок окна браузера. Название записывается между тегами `<TITLE>` и `</TITLE>` и представляет собой строку текста. Длина этой строки может быть любой, но рекомендуется делать ее не больше 60 символов. Элемент TITLE должен находиться только в разделе HEAD.

*



*

* 2.2. РАЗДЕЛ ДОКУМЕНТА BODY

*

- * В этом разделе документа располагается та информация, которая отображается в окне браузера. Раздел BODY должен начинаться тегом `<BODY>` и завершаться тегом `</BODY>`, между которыми располагаются элементы HTML, из которых и состоит содержание документа.

2. СТРУКТУРА HTML-ДОКУМЕНТА

2. СТРУКТУРА HTML-ДОКУМЕНТА

* 2.2.1. Спецификация элемента BODY

*

* Тег <BODY> имеет ряд атрибутов, определяющих внешний вид документа. Ниже приводится спецификация тега <BODY>.

* <BODY

* TEXT="цвет текста"

* BGCOLOR="цвет фона"

* BACKGROUND="адрес фонового рисунка"

* LINK="цвет непосещенной гиперссылки"

* VLINK="цвет посещенной гиперссылки"

* ALINK="цвет активной гиперссылки"

* >

* Атрибут TEXT задает цвет шрифта для всего документа в формате RGB или в символьной нотации. По умолчанию (если не указан этот атрибут) используются настройки браузера.

* Атрибут BGCOLOR задает цвет фона окна браузера документа в формате RGB или в символьной нотации. По умолчанию используются настройки браузера.

* Атрибут BACKGROUND позволяет указать адрес и имя рисунка, используемого в качестве фона. Этот рисунок будет размножен и распределен на заднем плане документа.

* Атрибуты LINK, VLINK и ALINK задают цвета гиперссылок в формате RGB или в символьной нотации. По умолчанию используются настройки браузера. Непосещенная гиперссылка - гиперссылка, которая еще не использовалась для перехода к другому документу. Посещенная гиперссылка - гиперссылка, которая уже использовалась для перехода к другому документу. Активная гиперссылка - гиперссылка на документ, к которому в данный момент происходит переход.

*

* 2.2.2. Советы по использованию атрибутов тега BODY

*

* · Если вы указываете хотя бы один цвет в теге BODY, то укажите и остальные. Это связано с тем, что пользователь может установить настройки цветов своего браузера как ему удобней. Указание только одного цвета может привести к ситуации, что у некоторых пользователей текст

* 2.3. ПРИМЕРЫ

* 2.3.1. Пример простого HTML-документа

*

* <HTML>

* <HEAD>

* <TITLE>Приветствие</TITLE>

* </HEAD>

* <BODY>

* <P>Добро пожаловать!</P>

* </BODY>

* </HTML>

*

Этот документ отобразится в браузере так:

*

* В этом примере обратите внимание на то, куда выводится браузером название документа в элементе TITLE.

*

* 2.3.2. Пример использования фонового рисунка

*

* <HTML>

* <HEAD>

* <TITLE>Приветствие</TITLE>

* </HEAD>

* <BODY BACKGROUND="smail.gif">

* <P>Добро пожаловать!</P>

* </BODY>

* </HTML>

*



2.

СТРУКТУРА HTML-ДОКУМЕНТА

* Для определения цвета различных элементов HTML-документа необходимо указать значение соответствующих атрибутов. Указывать значения этих атрибутов можно двумя способами:

* · определять цвет в формате RGB;

* · определять цвет, используя символьную нотацию

*

* 3.1. ФОРМАТ RGB

*

* Формат RGB - это система указания цвета, которая базируется на смешении трех основных цветов: красном (RED), зеленом (GREEN) и синем (BLUE). Итоговый цвет определяется цифрами в шестнадцатеричном коде. Для каждого цвета задается шестнадцатеричное значение в пределах от 0 до FF, что соответствует диапазону 0-255 в десятичном исчислении. Затем эти значения объединяются в одно число, перед которым ставится символ #. Например, число #800080 обозначает фиолетовый цвет. Указывая цвет в формате RGB, можно определить более шестнадцати миллионов цветовых оттенков.

*

* 3.2. СИМВОЛЬНАЯ НОТАЦИЯ

*

* Задание цвета в формате RGB имеет один недостаток - необходимо помнить совокупности цифр для указания цвета. Этого недостатка лишена символьная нотация. Можно указывать название цвета на английском языке. Но у этого способа указания цвета тоже есть недостаток - определено всего шестнадцать имен цветов.

* 3. ЦВЕТОВЫЕ СПЕЦИФИКАЦИИ

* Любые тексты, будь то школьные сочинения, заметка в газету или техническое описание устройства, имеют определенную структуру. Элементами такой структуры являются заголовки, подзаголовки, абзацы, списки и др.

* Разбиение всего текста на структурные элементы называется логическим форматированием. В HTML-документе логическое форматирование достигается с помощью специальных тегов.

*

*

* 4.1. АБЗАЦЫ

*

* Одним из первых правил составления любых документов является разбиение его текста на отдельные абзацы, выражающие законченную мысль. В HTML-документе разделение на абзацы производится с помощью специального тега <P>. Синтаксис этого тега таков:

* <P

* ALIGN="выравнивание">

* Атрибут ALIGN определяет способ выравнивания абзаца. Он может иметь следующие значения:

* · LEFT - текст выравнивается по левому краю окна браузера. Это значение используется по умолчанию, т.е. когда атрибут не указан.

* · CENTER - текст выравнивается по центру окна браузера.

* · RIGHT - текст выравнивается по правому краю окна браузера.

* Пример использования тега <P>:

* <HTML>

* <HEAD>

* <TITLE>Приветствие</TITLE>

* </HEAD>

* <BODY>

* <P>Добро пожаловать!</P>

* <P ALIGN="CENTER">Добро пожаловать!</P>

* <P ALIGN="RIGHT">Добро пожаловать!</P>

* </BODY>

* </HTML>



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

* 4.2. УПРАВЛЕНИЕ ПЕРЕВОДОМ СТРОКИ

*

* Так как браузер автоматически определяет места переноса строк, иногда возникают ситуации запретить перевод строки в каком-нибудь месте или, наоборот, принудительно сделать перевод строки в каком-то определенном месте. Для этого существуют особые теги, управляющие переводом строк.

* Когда необходимо сделать принудительный перевод строки, используют тег `
`. Этот тег не имеет атрибутов и закрывающего тега. Пример использования принудительного перевода строки:

* `<HTML>`

* `<HEAD>`

* `<TITLE>Приветствие</TITLE>`

* `</HEAD>`

* `<BODY>`

* `<P>Добро
пожаловать!</P>`

* `</BODY>`

* `</HTML>`



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

- * При использовании тега
 пустая строка не образуется, т.е. абзац не прерывается.
- * В некоторых случаях, наоборот, бывает необходимо сделать так, чтобы браузер не производил перевода строки. Например, не рекомендуется отрывать буквы инициалов от фамилии. В таких случаях тот участок текста, в котором нельзя переводить строку, следует поместить в элемент NOBR.
- * Пример:
- * <P>Это стихотворение написал <NOBR>А.С.
- * Пушкин</NOBR> - великий русский поэт.</P>
- * В браузере участок текста “А.С. Пушкин” всегда будет отображаться на одной строке.
- * Если получится так, что строка, расположенная в элементе NOBR, будет выходить за пределы окна браузера, то внизу окна появится горизонтальная полоса прокрутки.

* 4.3. ЗАГОЛОВКИ

- * Почти в каждом тексте используются заголовки для отдельных частей документа. Эти заголовки представляют собой фрагменты текста, которые выделяются на экране при отображении страницы браузером.
- * Для разметки заголовков используются теги <H1>, <H2>, <H3>, <H4>, <H5> и <H6>. Эти теги требуют соответствующего закрывающего тега. Заголовок с номером 1 является самым крупным (заголовок верхнего уровня), а с номером 6 - самым мелким. Теги заголовка нельзя использовать для выделения отдельных слов текста с целью увеличения их размера. При использовании тегов заголовков происходит вставка пустой строки до и после заголовка, поэтому тегов абзаца и перевода строки здесь не требуется.
- * Синтаксис тегов заголовков:

* <Hn

* ALIGN="выравнивание">

* Атрибут ALIGN определяет способ выравнивания заголовка. Он может иметь те же значения, что и аналогичный атрибут у тега абзаца.

* Пример использования разных заголовков:

* <HTML>

* <HEAD>

* <TITLE>Пример</TITLE>

* </HEAD>

* <BODY>

* <H1>Заголовок 1</H1>

* <H2 ALIGN="CENTER">Заголовок 2</H2>



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

* 4.4.2. Нумерованный список

* Нумерованные списки иногда называют упорядоченными. Списки данного типа представляют собой упорядоченную последовательность отдельных элементов. Отличием от маркированных списков является то, что в нумерованном списке перед каждым его элементом автоматически проставляется порядковый номер. Вид нумерации зависит от браузера и может задаваться атрибутами тегов списка. В остальном реализация нумерованного списка во многом похожа на реализацию маркированного списка.

* Для создания нумерованного списка следует использовать тег-контейнер , внутри которого располагаются все элементы списка.

* Каждый элемент списка должен начинаться тегом и заканчиваться тегом .

* Спецификация элемента OL:

* <OL

* TYPE="вид нумерации"

* START="начальная позиция">

* Атрибут TYPE задает вид нумерации, которой выделяются элементы списка. Он может иметь следующие значения:

* · A - маркеры в виде прописных латинских букв;

* · a - маркеры в виде строчных латинских букв;

* · I- маркеры в виде больших римских цифр;

* · i- маркеры в виде маленьких римских цифр;

* · 1- маркеры в виде арабских цифр, это значение используется по умолчанию.

* Атрибут START определяет позицию, с которой начинается нумерация списка. Используя этот атрибут, можно начать нумерацию, например, с цифры 5 или буквы E, в зависимости от вида нумерации. Значением атрибута START является число, вне зависимости от вида нумерации.

* Спецификация элемента LI для нумерованного списка:

* <LI

* TYPE="вид нумерации"

* VALUE="номер элемента">

* Атрибут TYPE задает вид нумерации, он может принимать такие же значения, что и одноименный атрибут элемента OL. По умолчанию значение этого атрибута - 1.

* Атрибут VALUE позволяет изменить номер данного элемента, при этом изменятся номера и последующих элементов.

* Пример использования нумерованного списка:

* <HTML>

* <HEAD>

* <TITLE>Пример списка</TITLE>

* </HEAD>

* <BODY>

*

* Города России по величине:

* Москва

* Санкт-Петербург

* Новосибирск

*

* </BODY>

* </HTML>



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

* 4.4.3. Список определений

*

* Списки определений, также называемые словарями определений специальных терминов, являются особым видом списков. В отличие от других типов списков каждый элемент списка определений всегда состоит из двух частей. В первой части элемента списка указывается определяемый термин, а во второй части - текст в форме словарной статьи, раскрывающий значение термина.

* Списки определений задаются с помощью тега-контейнера <DL>. Внутри него тегом <DT> отмечается определяемый термин, а тегом <DD> - абзац с его определением. Внутри элемента <DT> нельзя использовать абзацы (P) и заголовки (H1-H6), но их можно использовать внутри элемента <DD>. Атрибутов для элементов списка определений нет. В общем случае список определений записывается следующим образом:

* <DL>

* <DT>Термин</DT>

* <DD>Определение термина</DD>

* </DL>

* Пример использования списка определений:

* <HTML>

* <HEAD>

* <TITLE>Пример списка определений</TITLE>

* </HEAD>

* <BODY>

* <DL>

* <H2 ALIGN="CENTER">Состав Microsoft Office</H2>

* <DT>Microsoft Word</DT>

* <DD>Многофункциональный текстовый процессор</DD>

* <DT>Microsoft Excel</DT>

* <DD>Программа для работы с электронными



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

* 4.5. ФОРМАТИРОВАНИЕ ТЕКСТА

*

* В языке HTML предусмотрены специальные теги, предназначенные для форматирования текста. Они позволяют изменять вид шрифта, цвет, раз-мер и др.

* Чтобы отобразить текст полужирным шрифтом, используют тег . Например:

* <P>Это полужирный шрифт</P>

* Тег <I> отображает текст курсивом. Например:

* <P>Выделение <I>курсивом</I></P>

* Используя тег <TT>, можно отобразить текст шрифтом, в котором все буквы имеют одинаковую ширину. Это так называемый моноширинный шрифт. Например:

* <P>Это <TT>моноширинный</TT> шрифт</P>

* Для отображения текста подчеркнутым используется тег <U>. Например:

* <P>Пример <U>подчеркивания</U> текста</P>

* Тег <S> отображает текст, перечеркнутый горизонтальной линией. Например:

* <P>Пример <S>зачеркивания</S> текста</P>

* Тег <BIG> выводит текст шрифтом большего (чем непомеченная часть текста) размера. Например:

* <P>Шрифт <BIG>большого</BIG> размера</P>

* Тег <SMALL> выводит текст шрифтом меньшего размера. Например:

* <P>Шрифт <SMALL>меньшего</SMALL> размера</P>

* Тег <SUB> сдвигает текст ниже уровня строки и выводит его (если возможно) шрифтом меньшего размера. Удобно использовать для математических индексов. Например:

* <P>Шрифт _{нижнего} индекса</P>

* Тег <SUP> сдвигает текст выше уровня строки и выводит его (если возможно) шрифтом меньшего размера. Удобно использовать для задания степеней чисел в математике. Например:

* <P>Шрифт ^{верхнего} индекса</P>

* Теги форматирования могут быть вложенными друг в друга. При этом нужно внимательно следить, чтобы один контейнер находился целиком в другом контейнере. Например:

* <P>Этот текст <I>полужирный и

* курсивный</I></P>



4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

- * 4.5.1. Тег
- *
- * Тег позволяет изменить шрифт для блока текста. Этот тег имеет следующую спецификацию:
- * <FONT
- * FACE="тип шрифта"
- * COLOR="цвет шрифта"
- * SIZE="размер шрифта"
- * >
- * Атрибут FACE служит для указания типа шрифта, которым браузер будет выводить текст (если такой шрифт имеется на компьютере). Значением данного атрибута служит название шрифта, которое должно в точности совпадать с названием шрифта, имеющимся у пользователя. Если такой шрифт не найдется, то данное указание проигнорируется и будет использован шрифт, установленный по умолчанию.
- * Можно установить как один, так и несколько названий шрифтов, разделяя их запятыми. Тогда список шрифтов будет просматриваться слева направо так: если на компьютере пользователя нет шрифта, указанного в списке первым, то делается попытка найти следующий и т.д. Если ни одного шрифта найти не удалось, то будет использоваться шрифт, установленный браузером по умолчанию.
- * Атрибут COLOR устанавливает цвет шрифта. Значение этого атрибута может быть указано в формате RGB или символьной нотацией.
- * Атрибут SIZE служит для указания размера шрифта. Указывать размер шрифта можно двумя способами: абсолютной величиной или относительной величиной. При указании размера абсолютной величиной значением атрибута является число от 1 до 7. 1 - самый маленький шрифт, 7 - самый большой. При указании размера относительной величиной значением атрибута является число со знаком + или -. В данном случае шрифт будет увеличен (+) или уменьшен (-) от размера, принятого по умолчанию.
- * Пример использования тега :
- * <HTML>
- * <HEAD>
- * <TITLE>Пример изменения шрифта</TITLE>
- * </HEAD>
- * <BODY>
- * <P>Шрифт по умолчанию

- * зеленый шрифт

- * другая форма шрифта

- * размер шрифта - 6

- * размер увеличен на 1</P>
- * </BODY>
- * </HTML>

4. ВЫВОД ТЕКСТОВОЙ ИНФОРМАЦИИ

- * Как уже было сказано, гиперссылки - переходы к другим документам. Они являются основой HTML. Гиперссылки можно использовать для перехода не только к другим HTML-документам, но и другим объектам, которые можно разместить на компьютере, например, к видео- и аудиофайлам, архивам, рисункам и т.п.
- * Каждая ссылка состоит из двух частей. Первая - это то, что отображается браузером. Она называется **указателем ссылки**. Вторая часть - адресная, содержащая адрес объекта, к которому должен происходить переход. Эту часть называют **URL** (универсальный идентификатор ресурсов).
- * Когда вы щелкаете мышью по указателю ссылки, браузер загружает документ, адрес которого указан в URL.
- * Указателем ссылки может быть слово, группа слов или рисунок. Если указатель текстовый, то он обычно отображается браузером подчеркнутым синим шрифтом. При наведении курсора мыши на указатель курсор принимает форму руки, указывая, что это ссылка и можно произвести переход. Если указатель графический, внешне он не отличается от других рисунков, но при наведении курсора мыши на такой указатель, курсор также принимает форму руки.
- *
- * **5.1. УНИВЕРСАЛЬНЫЙ ИДЕНТИФИКАТОР РЕСУРСОВ URL**
- *
- * По своей сути URL - это адрес файла, к которому происходит переход. Указание адреса может быть относительным или абсолютным.
- * Когда в URL указывается полный путь к файлу, независимо от того, где этот файл находится (в Интернете или на жестком диске компьютера), то это абсолютное указание. Например: `http://www.mysite.ru/page.htm` - абсолютный адрес документа, находящегося в Интернете, `c:\web\doc1.htm` - абсолютный адрес документа на диске с.
- * Если в URL указывается не полный путь, а путь относительно адреса документа, в котором находится ссылка, то это относительное указание. Например, браузер отображает документ, абсолютный адрес которого `c:\web\doc1.htm`, в этом документе имеется ссылка с адресом `pict/ris1.jpg`, то это означает, что на самом деле ссылка будет на документ по адресу `c:\web\pict\ris1.jpg`.
- * Когда надо сослаться на файл, находящийся на другом компьютере, тогда следует пользоваться абсолютным указанием, а если сослаться на файлы, находящиеся на том же компьютере, где и документ, содержащий ссылку, то лучше использовать относительное указание.

5. ГИПЕРССЫЛКИ

* 5.2. ПРАВИЛА ЗАПИСИ ССЫЛОК

*

* Для организации ссылки необходимо указать браузеру, что является указателем ссылки, а также определить адрес документа, на который происходит ссылка. Оба этих действия выполняются с помощью тега <A>.

* Тег <A> имеет следующую спецификацию:

* <A

* HREF="URL-адрес"

* NAME="имя ссылки"

* TARGET="объект для вывода"

* >

* Атрибут HREF используется для задания адреса файла, к которому производится переход. Значением этого атрибута является текстовая строка, содержащая абсолютный или относительный URL-адрес.

* Атрибут NAME предназначен для задания ссылке имени. Значением этого атрибута является короткая текстовая строка. Этот атрибут используется для ссылок внутри одного HTML-документа.

* Атрибут TARGET позволяет определить, куда будет выводиться документ, на который происходит переход. Этот атрибут может иметь значение _blank - это означает, что документ будет выводиться в новом окне браузера.

* Пример ссылки:

* Документ 1

* Браузер отобразит эту строку так:

* Документ 1

* При нажатии мышью на этой строке браузер загрузит и отобразит файл doc1.htm, т.е. "Документ 1" - это указатель ссылки, а "doc1.htm" - URL-адрес.

*

* 5. ГИПЕРССЫЛКИ

- * Кроме ссылок на другие документы, часто бывает полезно включить ссылки на различные части текущего документа. Например, большой документ читается лучше, если он имеет оглавление со ссылками на соответствующие разделы.
- * Для построения внутренней ссылки сначала нужно создать указатель, определяющий место назначения. Для этого в месте, куда потом будет производиться ссылка, надо поместить тег <A> с атрибутом NAME, определив этим атрибутом имя указателя.
Например:
 - *
- * Обратите внимание, что в этом примере отсутствует содержимое тега <A>. Обычно так и делают, поскольку здесь нет необходимости как-то выделять текст, а требуется лишь указать местоположение.
- * После того как место назначения определено, можно приступить к созданию ссылки на него. Для этого в атрибуте HREF тега <A> помещается имя ссылки с префиксом #, говорящим о том, что это внутренняя ссылка.
Например:
 - * Глава 5
- * Теперь, если пользователь щелкнет кнопкой мыши на словах “Глава 5”, то браузер выведет соответствующую часть документа в окне просмотра.
- * Можно совмещать внутренние ссылки со ссылками на другие документы. Например:
 - * Глава 5 Документа 1
- * При нажатии на эту ссылку браузер откроет файл doc1.htm, найдет в этом файле указатель glava5 и выведет в окне просмотра соответствующую информацию.

* 5. ГИПЕРССЫЛКИ

- * Одним из достоинств HTML-документа является возможность использования графических элементов в оформлении. Можно выделить три элемента, чаще всего используемых в HTML-документах: горизонтальные линии, таблицы и рисунки.

*

* 6.1. ГОРИЗОНТАЛЬНЫЕ ЛИНИИ

*

- * Горизонтальные линии визуально подчеркивают законченность той или иной области документа. Сейчас часто используют рельефную, вдавленную линию, чтобы обозначить “объемность” документа.
- * Тег `<HR>` позволяет провести рельефную горизонтальную линию в окне большинства браузеров. Этот тег не является контейнером, поэтому не требует закрывающего тега. До и после линии автоматически вставляется пустая строка. Спецификация тега `<HR>`:

* `<HR`

* `ALIGN="выравнивание"`

* `WIDTH="длина линии"`

* `SIZE="толщина линии"`

* `NOSHADA`

* `>`

* Атрибут `ALIGN` определяет способ выравнивания линии. Он может иметь следующие значения:

- * `LEFT` - линия выравнивается по левому краю окна браузера. Это значение используется по умолчанию.
- * `CENTER` - линия выравнивается по центру окна браузера.
- * `RIGHT` - линия выравнивается по правому краю окна браузера.

* Атрибут `WIDTH` задает длину линии. Значением данного атрибута является число. Это число означает длину линии в пикселях. Если после числа стоит знак `%`, то это означает длину в процентах от ширины окна. Например:

* `<HR WIDTH="400">` - линия длиной 400 пикселей.

* `<HR WIDTH="50%">` - линия длиной 50 процентов от ширины окна.

* Атрибут `SIZE` задает толщину линии. Значением этого атрибута является число. Это число означает толщину линии в пикселях.

* Атрибут `NOSHADA` отменяет “трехмерность” линии.

6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ

* 6.2. ТАБЛИЦЫ

*

* Одним из наиболее мощных и широко применяемых в HTML средств являются таблицы. Они используются не только традиционно как метод представления данных, но и как средство форматирования Web-страниц. Документ HTML может содержать произвольное число таблиц, причем допускается вложенность таблиц друг в друга.

* Каждая таблица начинается тегом `<TABLE>` и заканчивается тегом `</TABLE>`. Внутри этой пары тегов располагается описание содержимого таблицы. Любая таблица состоит из одной или нескольких строк, в которых задаются данные для отдельных ячеек.

* Каждая строка начинается тегом `<TR>` и заканчивается тегом `</TR>`. Отдельная ячейка в строке обрамляется парой тегов `<TD>` и `</TD>` или `<TH>` и `</TH>`. Тег `<TH>` используется для ячеек заголовка таблицы, а `<TD>` - для ячеек данных. Отличие этих тегов в том, что в заголовке по умолчанию используется полужирный шрифт, а для данных - обычный.

* Теги `<TD>` и `<TH>` не могут появляться вне описания строки таблицы `<TR>`.

* Пример таблицы:

* `<HTML>`

* `<HEAD>`

* `<TITLE>Пример таблицы</TITLE>`

* `</HEAD>`

* `<BODY>`

* `<TABLE>`

* `<TR>`

* `<TD>Ячейка 1</TD>`

* `<TD>Ячейка 2</TD>`

* `</TR>`

* `<TR>`

* `<TD>Ячейка 3</TD>`

* `<TD>Ячейка 4</TD>`

* `</TR>`

* `</TABLE>`

* `</BODY>`

* `</HTML>`

* 6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ

- * Спецификация тега <TABLE>:
- * <TABLE
- * ALIGN="выравнивание"
- * BORDER="толщина рамки"
- * CELLPADDING="расстояние"
- * CELLSPACING="расстояние"
- * HEIGHT="высота"
- * VALIGN="вертикальное выравнивание"
- * WIDTH="ширина"
- * >
- * Атрибут ALIGN определяет выравнивание таблицы в окне просмотра браузера. Он может иметь одно из двух значений: LEFT (по левому краю) и RIGHT (по правому краю). По умолчанию - LEFT.
- * Атрибут BORDER управляет толщиной рамки. Значением этого атрибута является число. Это число определяет толщину рамки таблицы в пикселях. По умолчанию толщина рамки - 1.
- * Атрибут CELLPADDING определяет расстояние в пикселях между рамкой и содержимым ячейки. По умолчанию - 1.
- * Атрибут CELLSPACING определяет расстояние в пикселях между ячейками таблицы. По умолчанию - 2.
- * Атрибут HEIGHT определяет высоту таблицы в пикселях.
- * Атрибут VALIGN определяет вертикальное выравнивание содержимого таблицы. Он может иметь следующие значения: TOP (по верхнему краю), MIDDLE(посередине) и BOTTOM (по нижнему краю). По умолчанию - MIDDLE.
- * Атрибут WIDTH определяет ширину таблицы в пикселях или процентах от ширины окна браузера.
- *
- * Спецификация тега <TR>
- * <TR
- * ALIGN="выравнивание"
- * BGCOLOR="цвет фона"
- * VALIGN="вертикальное выравнивание"
- * >
- * Атрибут ALIGN определяет выравнивание содержимого всех ячеек строки. Он может иметь одно из трех значений: LEFT (по левому краю), RIGHT (по правому краю) и CENTER (по центру). По умолчанию - CENTER.
- * Атрибут BGCOLOR определяет цвет фона для всех ячеек строки. Его значение можно указывать в символьной нотации или в формате RGB.
- * Атрибут VALIGN определяет вертикальное выравнивание содержимого всех ячеек строки. Он может иметь следующие значения: TOP (по верхнему краю),MIDDLE (посередине) и BOTTOM (по нижнему краю). По умолчанию - MIDDLE.
- *
- * Спецификация тега <TD>
- * <TD
- * ALIGN="выравнивание"
- * BGCOLOR="цвет фона"
- * COLSPAN="количество ячеек"
- * HEIGHT="высота ячейки"
- * ROWSPAN=" количество ячеек "
- * VALIGN="вертикальное выравнивание"
- * WIDTH="ширина ячейки"

6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ

* 6.3. РИСУНКИ

*

* Без иллюстраций документ скучен, вял и однообразен. HTML позволяет использовать рисунки в формате JPG и GIF для оформления HTML-документов. Для вставки рисунков используется тег ****. Спецификация тега ****:

*

* **<IMG**

* SRC="адрес рисунка"

* ALIGN="выравнивание"

* HEIGHT="высота рисунка"

* WIDTH="ширина рисунка"

* BORDER="толщина рамки"

* **>**

*

* 6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ

- * Атрибут SRC определяет URL-адрес рисунка, который будет отображаться браузером.
- * Атрибут ALIGN определяет способ выравнивания рисунка. Он может иметь следующие значения:
 - * · TOP - рисунок выравнивается по верхнему краю текущей строки.
 - * · MIDDLE - рисунок выравнивается серединой по текущей строке.
 - * · BOTTOM - рисунок выравнивается по нижнему краю текущей строки.
 - * · LEFT - рисунок прижимается к левому краю окна браузера и обтекается текстом.
 - * · RIGHT - рисунок прижимается к правому краю окна браузера и обтекается текстом.
- * Атрибут HEIGHT определяет высоту рисунка в пикселях.
- * Атрибут WIDTH определяет ширину рисунка в пикселях.
- * Используя атрибуты HEIGHT и WIDTH можно увеличивать или уменьшать рисунок. Если указать только один из этих атрибутов, то рисунок будет увеличен или уменьшен пропорционально и по ширине, и по высоте.
- * Атрибут BORDER позволяет задавать рамку вокруг рисунка. Значение этого атрибута - толщина рамки в пикселях. По умолчанию - 1.
- * Пример выравнивания рисунков:

```
<HTML>
<HEAD>
<TITLE>Пример выравнивания</TITLE>
</HEAD>
<BODY>
<P>Выравнивание<IMG SRC="ris1.jpg" ALIGN="TOP">по
верхнему краю</P>
<P>Выравнивание<IMG SRC="ris1.jpg" ALIGN="BOTTOM">
по нижнему краю</P>
<P>Выравнивание<IMG SRC="ris1.jpg" ALIGN="MIDDLE">
по середине</P>
</BODY>
</HTML>
```

* 6. ГРАФИЧЕСКИЕ ЭЛЕМЕНТЫ

*

* Здесь были рассмотрены основные элементы языка HTML. На самом деле возможности данного языка неизмеримо больше. Для более детального изучения HTML рекомендуется обратиться к соответствующей литературе.

* 7. ЗАКЛЮЧЕНИЕ

*

*1. *Александровский А. Д.* Создание Web-страниц с использованием FrontPage 98 и JavaScript. - М.: ДМК, 1998.

*2. *Матросов А., Сергеев А., Чаунин М.* HTML 4.0. - СПб.: БХВ-Санкт-Петербург, 2000.

*3. Материалы из Интернета.

*** ЛИТЕРАТУРА**

