

Основы веб-конструирования

Свои странички в Интернете имеют все: государственные учреждения, большие компании, мелкие предприниматели, магазины, библиотеки, школы, детские сады.

***Web-пространство** - это мировая совокупность сайтов. Сайты расположены на сервере, серверы соединены между собой при помощи каналов связи.*

***Сайт** - это совокупность гипертекстовых страниц (чаще имеющая древовидную структуру) для представления предприятия, фирмы (корпоративные сайты), для представления школ (школьные сайты), частных лиц (личные сайты). Сайт может осуществлять коммерцию (интернет-магазины), поставлять информационные услуги (сайты для поиска информации, библиотеки, справочные системы, списки рассылки, обучение). В современном мире сайт становится визитной карточкой, как организации, так и частного лица.*

Важной особенностью этой обширнейшей информационной базы является её постоянное обновление и создание новых страниц. Среди многообразия выбора программ для создания веб-страниц (Microsoft Word, Microsoft FrontPage, HomeSite 5, CoffeeCup HTML Editor 8.5, NetObjects Fusion 5.0, Macromedia Dreamweaver др.).

*Гипертекстовые страницы, которые составляют сайт, записываются на специальном языке **HTML** (Hyper Text Markup Language). Этот язык не является, строго говоря, языком программирования, хотя он и интерпретируется браузером. Средства HTML служат для указаний браузеру, в каком виде выводить на экран текст и графику, а также позволяют записывать на страницах гиперссылки.*

В настоящее время широко используются два типа редакторов HTML:

- 1. Редакторы типа "что видишь, то и получишь" (Microsoft Front Page и др.). Пользователь не видит "внутренностей" документа, с*

которым он работает, точно так же, как при работе с текстовым процессором типа Microsoft Word. Существует довольно много конвертеров, способных преобразовывать документы, созданные в Microsoft Word в HTML-документы.

2. Редакторы собственно HTML-текстов (Macromedia Dreamweaver и многие другие). В процессе работы пользователь видит внутреннее содержание HTML-файла и может изменять его либо вручную, либо вызывая команды меню для вставки определенных элементов HTML. Работа с таким редактором очень похожа на работу с интерактивной средой программирования типа Microsoft Visual Basic или Borland Delphi.

Кроме языка разметки гипертекстовых документов HTML для создания веб-сайтов используются другие языки программирования, например Java Script и PHP. Они используются, как правило, для создания отдельных элементов веб-страницы, например, выпадающих списков, динамических меню, часов, календарей и т.п.

Приступая к созданию сайта необходимо выделить основные этапы создания ресурса. Именно в момент «конструирования» происходят основные ошибки, в результате которых не достигается поставленная задача. Разработка сайта - это процесс вполне управляемый, а предварительное планирование - это крайне необходимое дело.

1. Разработка тематики сайта, его целей и задач.
2. Разработка структуры и дизайна сайта.
3. Заполнение материалом. Необходимо предварительно подготовить первоначальный пакет документов, новостей, статей, фотографий и рисунков.
4. Публикация сайта в Интернете. Суть этого этапа заключается в двух вещах. Первая - это имя сайта, иначе **домен**. Вторая - размещение сайта, иначе **хостинг**. Под доменом понимают то имя ресурса, которое и будет адресом сайта. Под хостингом понимается адрес физического размещения сайта.

5. *Поддержка сайта.* Кроме наполнения сайта необходимым материалом нужно уделять серьезное внимание и постоянно поддерживать ресурс, наполняя его самыми свежими новостями.

В наше время быстрого развития Интернета и увеличения количества сайтов в сети, первостепенное значение в привлечении посетителей на сайт имеет количество и качество актуальной информации. По управлению контентом (содержанием сайта) можно выделить три основных типа сайта: статический, динамический и флеш сайты. Конечно, очень часто создаются сайты, в которых есть разделы всех трех типов.

Статический html-сайт разработан с использованием стандартной html-технологии. Страницы сайта хранятся в виде html-кода в файловой системе сервера. Естественно, на таком сайте могут присутствовать флеш-заставки, ролики, анимация и т.п. Но для редактирования информации и обновления такого сайта необходимы серьезные затраты и постоянное привлечение квалифицированных веб-дизайнеров.

Основное свойство *динамических сайтов* - гибкость, подвижность информационной системы в подборе и представлении информации. Страницы сайта формируются сервером из различных блоков в соответствии с запрошенной посетителем страницей. В файловой системе сервера лежат исполняемые скрипты (программы). Весь контент (информация) сайта хранится в базе данных, которая часто физически расположена на другом сервере, недоступном из Интернета. Динамический сайт может содержать флеш-заставки, ролики, анимацию, а также отдельные статические страницы.

Флеш-сайт - это хорошо анимированные картинки или интерактивные мультики. В файловой системе сервера лежат статические html-страницы и флеш-ролики. При просмотре такого сайта на компьютер пользователя подгружаются флеш-ролики и, в соответствии с записанной в таком ролике программой, формируют для пользователя на экране картинку. Такая картинка, как правило, содержит интерактивные элементы (ссылки) при щелчке, на которые выводятся другие картинки или текст.

Основные понятия языка HTML. Структура HTML-документа

Все стандартные браузеры для сети Интернет используют способы представления текстов, основанные на языке HTML.

HTML (Hyper Text Markup Language) - это язык разметки гипертекста. Этот язык «понимают» все компьютеры, он довольно прост, но при этом имеет достаточные выразительные средства для удобного описания разных типов документов. Язык позволяет хранить текст в «чистом» виде (не кодируя его), что делает возможным просмотр его с помощью обычных текстовых редакторов. Документ HTML представляет собой обычный текстовый файл, содержащий конструкции языка HTML, который можно создавать в обычных текстовых редакторах (программа Блокнот, Microsoft Word), а затем сохранять созданные файлы с расширением .htm или .html.

Команды языка HTML задаются между специальными символами <...>, и называются тегами (tag). Между ними находятся коды, которые распознает браузер.

Тег - это инструкция браузеру, указывающая способ отображения текста. Теги позволяют управлять представлением информации на экране при отображении HTML-документов специальными программами - браузерами. (Например, Microsoft Internet Explorer или Netscape Navigator)

Приняв информацию, браузер компоует элементы документа в соответствии с тем, что задано командами HTML, и показывает результат на экране с учетом предварительных настроек, касающихся размеров окна браузера, используемой цветовой схемы, установленных цветов и т.д.

Теги могут быть двух видов:

1) **одноэлементный тег**: `<...>`; его достаточно просто вставить в текст для того, чтобы совершить какое-либо действие;

2) **парный тег**: `<...>...</...>`; он влияет на текст, с того места, где употреблен, до того места, где указан признак окончания его действия. Признаком завершения команды служит тот же самый тег, только начинающийся с символа `" / "`:

Типичный HTML - документ имеет следующую структуру:

`<html>`

`<head>`

`<title> описание заголовка </title>`

`</head>`

`<body>`

Текст документа

`</body>`

`</html>`

`<html>...</html>` - границы HTML-программы. Любой HTML-документ должен начинаться (и заканчиваться) тегом `<html>...</html>`. Этот тег указывает на то, что данный документ содержит в себе HTML-текст.

Веб-документ разбивается на две логические части: заголовок и содержание.

`<head>...</head>` - заголовок HTML-программы. Заголовок также называют головной частью программы, он содержит справочную информацию о странице, которая не отображается браузером, а также название документа.

В блоке заголовка размещается тег `<title> текст </title>`. Текст, указанный в этом теге отображается в заголовке окна браузера.

В HTML-документе можно разместить комментарии, которые браузером не отображаются: `<!-- Комментарии -->`.

Основное содержание страницы помещается в тег `<body>...</body>`. Его также называют телом программы. Тег `<body>` определяет внешний вид всей веб-страницы, в то время как отдельные ее элементы, например заголовки, таблицы, могут иметь свое особенное оформление. Теги языка HTML могут содержать **атрибуты**, которые являются параметрами

или свойствами элементов разметки документа. Правило записи атрибутов в теге следующее: **<тег атрибут1=значение атрибут2=значение>**.

Назначение атрибутов тега **<body>**:

bgcolor = цвет - цвет фона страницы;

text = цвет - цвет текста на странице;

link = цвет - цвет текста гиперссылок;

vlink = цвет - цвет использованных гиперссылок.

Значение цвета в атрибутах HTML-языка может задаваться несколькими способами: названием ("red"); кодом RGB (например "#FFFFFF"); в десятичном виде (например "255, 0, 0"); в процентах (например "30%, 40%, 30%").

Оформление HTML-документа. Текст. Таблицы

Теги оформления текста

- **Теги абзаца и перевода строки:**

<P> - новый абзац. Отделяет и добавляет пустую строку, которая зрительно выделяет абзац.

**
** - переход на новую строку. Отделяет строку от последующего текста или графики.

- **Теги выделения фрагментов текста** позволяют управлять отображением отдельных символов и слов.

<I> - текст курсивом

**** - полужирный шрифт

<U> - подчеркнутый шрифт

<Sup> - ^{верхний} индекс

<Sub> - _{нижний} индекс

Возможно использование комбинированных тегов по следующему правилу:

<Тег1><Тег2>...</Тег2></Тег1>

- **Использование стилей заголовка:** Тег **<Hn>** - заголовки документа, где *n* принимает значения уровня от 1(самый большой) до 6(самый маленький).

- Для выравнивания текста используется атрибут *Align=*, который встраивается в теги абзаца или заголовка, имеет одно из трех значений:
Align = center - выравнивание по центру;
Align = left - выравнивание по левому краю;
Align = right - выравнивание по правому краю.
- *Задание размера текущего шрифта:* Тэг шрифта ** позволяет задавать размер текущего шрифта в отдельных местах текста. Диапазон установки текущего шрифта от 1 до 7. ** и **
- С помощью атрибута *Face* тега *FONT* можно менять стиль написания (гарнитуру) шрифта, например: **
- Для изменения цвета шрифта в тэге ** можно использовать атрибут *Color=цвет*.
- *Теги для формирования списков:*
** и ** - показывают начало и конец нумерованного списка.
 Атрибуты тега *OL*: *TYPE* - вид счетчика:
A - большие латинские буквы,
a - маленькие латинские буквы,
I - большие римские цифры,
i - маленькие римские цифры,
1 - обычные цифры.
** и ** - начало и конец маркированного списка.
** Элемент списка

Теги оформления таблиц

Таблицы представляют собой особую часть HTML-документа. Данные в ней организованы в виде прямоугольной сетки, состоящей из вертикальных столбцов и горизонтальных рядов. Каждая клетка таблицы является ячейкой.

Ячейки могут содержать в себе текст, графику или другую таблицу.

Таблица состоит из трех основных частей: название таблицы, заголовки столбцов, ячейки.

Таблица заполняется горизонтальными рядами ячейка за ячейкой слева направо. Заполнение начинается с левого верхнего угла и заканчивается правым нижним. Каждая ячейка должна быть заполнена. Для создания пустых ячеек используются пробелы.

<i>Тег</i>	<i>Форма записи</i>	<i>Примечание</i>
<i>Table</i>	<code><table>текст</table></code>	<i>объявление таблиц</i>
<i>Tr</i>	<code><tr>текст</tr></code>	<i>тег строки</i>
<i>Td</i>	<code><td>текст</td></code>	<i>тег данны</i>

атрибуты тега <table>

<i>Атрибу т</i>	<i>Форма записи</i>	<i>Примечание</i>
<i>Border</i>	<code><table border=x></code>	<i>задает рамку вокруг таблицы, x-число (0-без рамки)</i>
<i>Width</i>	<code><table width=xx%></code>	<i>задает ширину таблицы как xx% от ширины страницы или как xx пикселей</i>
<i>Bgcolor</i>	<code><table bgcolor="#rrggbb"></code>	<i>задает цвет фона таблицы</i>

атрибуты тегов <td> и <tr>

<i>Атрибу т</i>	<i>Форма записи</i>	<i>Примечание</i>
---------------------	---------------------	-------------------

<i>Align</i>	<code><td align=x></code>	<i>выравнивание по горизонтали (right, left, center)</i>
<i>Valign</i>	<code><td valign=x></code>	<i>выравнивание по вертикали (top, middle, bottom, baseline)</i>
<i>Bgcolor</i> <i>r</i>	<code><td bgcolor="#rrggbb"></code>	<i>задает цвет фона ячейки</i>

Вставка графических изображений, гиперссылок. Подготовка изображений для сети Интернет

Размещение графики

Тэг `` позволяет вставить изображение в документ в форматах: gif, jpg, png. Изображение появится в том месте документа, в котором записан этот тэг.

Тэг `` имеет атрибуты, которые можно задавать дополнительно. Они могут располагаться где угодно в тэге после кода `img`.

Атрибуты изображения

<i>Атрибу т</i>	<i>Формат</i>	<i>Описание</i>
<i>alt</i>	<code></code>	<i>Если браузер не воспринимает изображение, вместо него появляется заменяющий текст</i>
<i>border</i>	<code></code>	<i>Задаёт толщину рамки вокруг изображения (в пикселях)</i>

<i>align</i>	<code></code>	<i>Выравнивает изображение относительно текста: по верхней части изображения - top, по нижней - bottom, по средней - middle</i>
<i>height</i>	<code></code>	<i>Задаёт вертикальный размер изображения внутри окна браузера</i>
<i>width</i>	<code></code>	<i>Задаёт горизонтальный размер изображения внутри окна браузера</i>
<i>vspace</i>	<code></code>	<i>Добавляет верхнее и нижнее пустые поля</i>
<i>hspace</i>	<code></code>	<i>Добавляет левое и правое пустые поля</i>

Фоновое изображение графики на Web-странице

Графика, используемая в качестве фоновой, задается в тэге <body> в начале файла html.

Например. <body background="bgr.gif" text="#330066"></body>

Построение гипертекстовых связей

Ссылки позволяют при помощи нажатия кнопки мыши быстро переходить от одного документа к другому не задумываясь, где находится этот документ. В качестве ссылки можно использовать текст или графику. Браузер выделяет (обычно цветом и/или подчеркиванием) слова, являющиеся ссылками. Цвет выделения ссылок устанавливается в тэге Body:

Link - устанавливает цвет выделения ссылок.

Vlink - устанавливает цвет выделения ссылок, на которых уже побывали.

Alink - устанавливает цвет активной ссылки.

Возможны ссылки:

- на удаленный HTML-файл;
- на некоторую точку в этом же документе;
- на любой файл, не являющийся HTML-документом.

Тег `<A>...` создает гиперссылку. Атрибуты: *Href* - имя документа (URL-адрес), к которому осуществляется переход; *Title* задает текст, который выскочит, если задержать курсор мыши на указателе; *Target* указывает, куда должна загружаться вызываемая страничка. Если указать *Target=_blank*, то браузер загрузит вызываемую страничку в новое окно. Если этот атрибут опустить, то новая страничка загрузится в текущее окно.

Ссылки в пределах одного документа.

Такие ссылки требуют наличие двух частей: метки и самой ссылки. Метка определяет точку, к которой происходит переход по ссылке.

Ссылка: `Понедельник`

Перед именем метки (nn) указывающей, куда производится ссылка, ставится символ #. Между символами > и < располагается текст (Понедельник), на котором производится щелчок для перехода по ссылке.

Метка: `Понедельник`

Ссылки на другой HTML-документ или ресурс Интернет.

Ссылки позволяют щелчком по выделенному слову или фразе перейти к другому HTML-документу или ресурсу.

Ссылка на другой документ: `11 классы;`

Ссылка на ресурс: `ресурс ;`

Гиперсвязь с адресом электронной почты: `<address> person@firm.ru</address>.`

Графическая ссылка на другой HTML-документ.

В качестве ссылки можно использовать не только текст, но и графику.

Для этого достаточно заключить тег `img` в тег `A`.

Например, ``.

Бегающая строка

Тег `<marquee>текст</marquee>` - создает бегающую строку.

`<marquee direction=left>текст</marquee>` - если бегающую строку нужно направить справа налево.

`<marquee direction=right>текст</marquee>` - движение слева направо.

`scroll` - стандартное движение от правого края к левому; `slide` - надпись один раз пробегает от правого края к левому, где и остается; `alternate` - движение от правого края страницы к левому и обратно

`<marquee loop=n behavior=scroll>текст</marquee>` - ограничение числа циклов, значение `n` оператора `loop` указывает число повторений цикла.

`<marquee width=n>текст</marquee>` - указать ширину участка, занимаемого бегающей строкой, где `n` - ширина той части страницы, на которой расположена бегающая строка. Значение `n` указывается как в пикселях, так и в процентах от общей ширины видимой части страницы.

`<marquee scrollamount=n>текст</marquee>` - регулировка движения надписи по экрану. здесь `n` - число пикселей.

`<marquee scrolldelay=t>текст</marquee>` в данном случае переменная величина - время `t` - измеряется в миллисекундах (времени, спустя которое текст будет перерисован на экране заново).

`<fontsize=n><marquee> текст</marquee>` - возможность указывать величину шрифта текста в строке.

`<marquee bgcolor=n> текст </marquee>` - окрасить поверхность бегающей строки в какой-либо цвет.

`<marquee height=n>текст</marquee>` - указать высоту бегающей строки, задавая величину `n` в пикселях.

Пример. Использование гиперссылок, графики.

```
<html>
<head> <title> учебный файл html </title> </head>
<title> расписание занятий 11 классов </title>
</head>
<body background="15.jpeg" text="#330066" >
<p align=center>
<font color="#008080" size="7"><b> расписание </b></font><br>
<font size="6"> <i>занятий на вторник</i></font><br><br>

</p>
<font color="blue" size="4" face="courier"><b>
<A href="2.html">вторник</A>
<marquee direction=left>расписание</marquee>
</body>
</html>
```

Подготовка изображений для сети Интернет

*При подготовке изображений для размещения на веб-страницах необходимо их оптимизировать, чтобы обеспечить минимальный размер файла. Получаемое с фотокамеры изображение, как правило, имеет большой размер. Такое изображение не удобно при просмотре в Интернете и долго загружается из-за большого размера. Кроме того, сайты, автоматически уменьшающие изображение, делают это зачастую некорректно или с большой потерей качества. Работа по оптимизации изображений для размещения в сети Интернет делится на два этапа: подготовка изображения в графическом редакторе и процесс сжатия. В редакторе Adobe PhotoShop для оптимизации изображений используются параметры, влияющие на степень возможного сжатия изображения. Для этого воспользуемся пунктом меню *Image/Image Size*. Основное, что следует помнить, что размеры изображения должны быть не меньше 600 и не больше 900 пикселей по большей стороне. В открывшемся диалоговом окне в разделе *Pixel Dimension* устанавливаем*

необходимое разрешение. При изменении любого параметра другой подстроится автоматически. Разрешение (*Resolution*) задаем стандартное для веб-72ppi, пропорции сохраняем (*Constrain Proportions*). Размер изображения в пикселях желательно задать минимально необходимый. Ориентироваться следует на самое распространенное разрешение экрана, которое в настоящее время составляет 1280x1024 (1280x800 для широкого экрана).

Полученное изображение будет иметь необходимый размер. Однако мы потеряли чёткость изображения, для этого воспользуемся стандартным фильтром *Smart Sharpen* из меню *Filter/Sharpen*. Устанавливаем параметры: Amount=75 (не более 90) Radius=0.3 (не более 1). Убедившись, что результат нас устраивает - нажимаем ОК.

При сохранении изображения для публикации в сети Интернет в Adobe Photoshop имеется функция *File/Save for Web & Devices*. Требования оптимизации для форматов jpeg, gif и png отличаются, что связано с разными алгоритмами сжатия. По этому в открывшемся окне обратим внимание на следующие параметры:

- а) тип сжатия - установим тип файла (первый раз нам предложат gif);
- б) качество сжатия - установим *High* или *Very High* или выставим ручную *Quality* - от 60 до 85 в зависимости от количества мелких объектов. Увеличение качества влияет напрямую на размер файла;
- в) установим сжатие *Optimized* или *Progressive* (*Progressive* позволит уменьшить размер файла);
- г) размер файла не должен превышать 400Кб.
- д) нажимаем *Save* и выбираем место сохранения.

Редактор визуального веб-конструирования. Публикация сайта

Программы для создания сайтов называются редакторами сайтов. Все их можно разделить на две категории: текстовые редакторы и визуальные редакторы. Для того чтобы пользоваться текстовыми редакторами потребуются знания языка HTML, вторые таких знаний не требуют.

Главное свойство и преимущество визуального редактора заключается в том, что программа сама пишет код веб-страниц, а пользователю остается только конструировать сайт, вставляя картинки, текст и другие элементы.

Прикладная программа FrontPage 2003 - это визуальный HTML-редактор для быстрого создания сайта. С помощью FrontPage, можно создавать структуру сайта, формировать страницы, добавлять интерактивные средства и загружать файлы на сервер в сети Интернет. FrontPage позволяет создавать сайт непосредственно в файловой системе компьютера пользователя, а затем, когда он будет готов, опубликовать его на веб-сервере.

Окно прикладной программы FrontPage

После запуска программы FrontPage (Пуск/Программы/FrontPage) на экране появится окно программы, в котором отображается новая страница.

Окно прикладной программы FrontPage состоит: из строки заголовка, строки меню, панелей инструментов редактирования и форматирования, кнопок быстрого выбора тега (для редактирования и ввода тегов), ярлычков вкладок выбора страниц (для перехода между страницами), рабочего окна, в котором отображается новая страница, панели рисования, кнопок режимов просмотра, индикатора ожидаемого времени загрузки страниц, индикатора размера страниц, области задач.

В программе FrontPage можно пользоваться одним из четырех режимов просмотра: ***Конструктор, Код, С разделением и Просмотр.***

В режиме ***Конструктора*** как в любом текстовом редакторе можно в визуальном режиме создавать, редактировать и форматировать

страницу, т.е. вводить текст, добавлять рисунки, таблицы. При этом теги языка HTML автоматически добавляются в фоновом режиме, но кодировка HTML на экране не отображается.

В режиме **Код** на экране будет отображаться вся кодировка и можно напрямую редактировать код HTML, а также вводить новые коды.

В режиме **С разделением** на экране отображается веб-страница одновременно в режиме **Код** и в режиме **Конструктор**.

Команды, которые предназначены для работы с веб-страницами и веб-узлами, расположены в меню **Вид** программы FrontPage:

- **Страница** - это режим просмотра и разработки страницы.
- **Папки** - отображается структура папок текущего сайта.
- **Удаленный узел** - узел, который находится на сервере сети Интернет.
- **Отчеты** - предоставляется сводка об веб - узле.
- **Переходы** - структура переходов между страницами сайта.
- **Гиперссылки** - открывает список ссылок текущей страницы.
- **Задачи** - открывает список задач для текущего сайта

Создание Web-страниц в программе FrontPage

Для создания новой пустой страницы необходимо выполнить команду **Файл/Создать/Пустая страница**. Далее необходимо осуществить разработку страницы, т.е. выполнить разметку (структуру) страницы, ввести текст, рисунки и т.д.

Создать новую страницу можно также на базе одного из шаблонов. Для этого необходимо выбрать команду **Другие шаблоны страниц** в области задач. На экране будет отображаться окно диалога **Шаблоны страниц**, в котором представлены различные шаблоны страниц по категориям.

Создать веб-страницу можно и на базе имеющихся, на компьютере веб-страниц. Для этого необходимо в области задач выбрать команду **Из имеющейся страницы** и в появившемся окне диалога выбрать требуемую

страницу. Далее можно произвести необходимые изменения на странице и сохранить ее под другим именем.

Создание сайта в программе FrontPage

Для создания нового пустого сайта необходимо выполнить команду **Файл/Создать/Другие шаблоны веб-узлов**. Откроется окно диалога, в котором необходимо выделить **Пустой веб-узел**.

Для создания пустой домашней страницы в новом веб-узле необходимо перейти в режим **Переходы/Новая страница**, в рабочем поле появится **домашняя страница**. Дважды щелкните на домашней странице левой кнопкой мыши, и она откроется в режиме конструктора для редактирования (страница - **index.htm**).

Далее можно осуществить разработку домашней страницы (создать разметку или структуру страницы, ввести текст, рисунки и т.д.), а затем к домашней странице можно добавить страницы в режиме **Переходы**. Страницы можно переименовать и изменить порядок их следования. На каждую страницу нужно ввести содержание (текст, таблицы, картинки и т.д.), в результате получим сайт.

Для создания сайта на основе шаблона выполните **Файл/Создать/Веб- пакеты**, откроется окно диалога. Выберите требуемый шаблон или мастера для создания нового сайта. Созданный сайт имеет разметку и дизайн, но в нем отсутствует контент (наполнение). Далее в страницы этого сайта необходимо ввести соответствующий текст, рисунки, скрипты, счетчики и другие элементы сайта.

- **Форматирования текста** в FrontPage осуществляется с помощью панели инструментов **Форматирование**, окон **Формат** и **Абзац** (аналогично Word). Отображение веб-страницы на экране зависит от разрешения монитора и настройки браузера, поэтому размер шрифта принято задавать в условных единицах от 1 до 7. Для создания текстовой бегающей строки необходимо выполнить:

Вставка/Веб-компонент/Бегающая строка. В окне *Свойства бегающей строки* вводится текст, задается направление и скорость движения, время задержки и другие параметры.

- *Размещение изображений* на веб-странице осуществляется *Вставка/Рисунок*. Задать параметры изображения на странице можно в меню *Формат/Положение*. В редакторе FrontPage предусмотрена возможность простейшей обработки изображений (коррекция яркости и контрастности, повороты, обрезка и т.д.) в меню *Рисунки*. Рисунки для веб-страницы должны быть заранее подготовлены и вставлены в папку *images*, причем имя рисунка должно состоять из символов латинского алфавита.
- На веб-странице можно вставлять таблицы в меню *Таблица* для структурирования информации. В диалоговом окне *Вставка таблицы* можно указать количество строк, столбцов, выравнивание, цвет фона ячеек, цвет и толщину границ, интервал между ячейками и другие параметры.
- Для создания *гиперссылки* необходимо выделить элемент в документе и в меню *Вставка/Гиперссылка* вызвать диалоговое окно и задать адрес перехода по данной ссылке. В качестве элементов привязки могут выступать слова, группы слов, изображения. Ссылки могут обеспечивать переход к документам, расположенным как на данном сайте (внутренние ссылки), так и на других сайтах (внешние ссылки).

Публикация сайта

Публикация сайта - это размещение сайта на сервере или локальном компьютере с возможностью вызова из сети (глобальной или локальной). Некоторые серверы бесплатно предоставляют такие услуги (например, *www.narod.ru*). Для публикации сайта, подготовленного в редакторе FrontPage, необходимо выполнить действия: *Файл/Опубликовать/Свойства удаленного веб-узла*.

Для размещения сайта на сервере narod.ru в строке *Расположение удаленного веб-узла* следует указать адрес ftp-узла *ftp://ftp.narod.ru*, ввести имя и пароль в окне *Требуется имя и пароль*. Для копирования всех файлов локального веб-узла на сервер необходимо нажать кнопку *Опубликовать веб-узел*.