**SVG графика в HTML5**

SVG (Scalable Vector Graphics – масштабируемая векторная графика) – стандарт векторной графики, разработанный консорциумом W3C.

SVG – это язык разметки для описания двухмерных графических приложений и изображений, входящий в подмножество расширяемого языка разметки XML. Сюда относится также ряд связанных графических скриптов.

SVG поддерживается всеми современными браузерами для ПК и мобильных телефонов. Некоторые функции, такие как SMIL анимации и SVG Fonts распространены не так широко.

Базовая поддержка SVG браузерами.

Последней версией полной спецификации является SVG 1.1.

SVG 2 находится в стадии разработки. В нее будут добавлены новые, простые в использовании функции для SVG, а также идут работы по более тесной интеграции с HTML, CSS и DOM.

**Достоинства SVG**

Графика в формате SVG создается с использованием математических формул, которые при изменении размера изображения можно скорректировать. Таким образом векторные изображения масштабируются лучше, чем растровые.

Размер векторной картинки обычно меньше, чем у сравнимых по качеству изображений в форматах JPEG, GIF или PNG.

SVG графика имеет текстовый формат, который можно и править в блокноте, и рисовать в графических векторных редакторах Adobe Illustrator, CorelDRAW.

Cкрипты и анимация в SVG позволяют создавать динамичную и интерактивную графику.

Текст в графике SVG является текстом, а не изображением, поэтому он индексируется поисковыми системами.

В SVG изображение можно добавить несколько ссылок.

К SVG формату можно подключать внешние таблицы стилей CSS, глобальные стили внутри контейнера <style>...</style> или добавлять внутренние стили с помощью атрибута style в тегах фигур и путей.

Вставка SVG на Web-страницу

SVG-изображение можно вставить на Web-страницу несколькими способами. Первый из них – это простая вставка SVG-кода на страницу (при большой картинке HTML-код страницы станет огромным и трудно читаемым). При других способах сначала надо сохранить SVG-код в файле с расширением .svg.

Итак, способы вставки SVG-изображения на Web-страницу:

прямая вставка кода в HTML-документ в контейнере <svg>...</svg>;

использование SVG-файла в качестве фонового изображения;

подключение SVG-файла в HTML-документ с помощью тегов img, embed, object и iframe;

подключение SVG-файла в PHP-документ с помощью функции include.

1. Прямая вставка SVG-кода в Web-документ

<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<!-- SVG-код -->

</svg>

2. Использование SVG-файла в качестве фонового изображения

<style>

.home {

display: block;

width: 200px;

height: 300px;

background: url(example.svg) 0 0;

background-size: contain;

}

</style>

<a href="/" class="home"></a>

3. Подключение SVG-файла с помощью тега img

<img src="example.svg">

4. Подключение SVG-файла с помощью тега embed

<embed src="example.svg" type="image/svg+xml">

5. Подключение SVG-файла с помощью тега object

<object data="example.svg" type="image/svg+xml"></object>

6. Подключение SVG-файла с помощью тега iframe

<iframe src="example.svg" width="200" height="300" style="border: none"></iframe>

7. Подключение SVG-файла с помощью функции include

<? include("example.svg"); ?>

**Система координат**

Размеры и координаты можно задавать в различных единицах (px, pt, pc, cm, mm, em, in). Если единицы измерения не указаны, то это пиксели.

Начало отсчета координат – это верхний левый угол экрана, т.е.

Базовые SVG-элементыline прямая линия

polyline ломанная линия

polygon многоугольник

rect прямоугольник

circle круг

ellipse эллипс

path сложная траектория

Общие атрибуты тегов SVG-элементовstroke Цвет линии

stroke-width Толщина линии

fill Цвет заливки (none – без заливки)

fill-opacity Прозрачность заливки (от 0 до 1)

fill-rule Правило заливки.

Возможные значение атрибута: nonzero сплошная заливка;

evenodd внутренняя часть фигуры не заливается.

style Стиль элемента

class Класс элемента

Прямая линия

Задается тегом <line>.

Атрибуты тега <line>x1 Координата начальной точки линии по оси X

y1 Координата начальной точки линии по оси Y

x2 Координата конечной точки линии по оси X

y2 Координата конечной точки линии по оси Y

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<line x1="5" y1="5" x2="500" y2="60" stroke="#b4241b" stroke-width="3" />

</svg>

РЕЗУЛЬТАТ:

Ломаная линия

Задается тегом <polyline>.

Атрибут тега <polyline>points Координаты точек ломанной линии парами x,y через пробел

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<polyline points="5,5 5,20, 150,20, 150,35, 295,35, 295,50, 440,50 440,65"

stroke="#b4241b" stroke-width="3" fill="none" />

</svg>

РЕЗУЛЬТАТ:

Многоугольник

Задается тегом <polygon>. Всегда выводит замкнутые фигуры, автоматически проводя линию из конца последнего отрезка в начало первого.

Атрибут тега <polygon>points Координаты точек многоугольника парами x,y через пробел

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<polygon points="88,10 56,90 136,42 40,42 120,90"

stroke="#b4241b" stroke-width="3" fill="#ffff00" fill-rule="nonzero"/>

<polygon points="188,10 156,90 236,42 140,42 220,90"

stroke="#b4241b" stroke-width="3" fill="#ffff00" fill-rule="evenodd"/>

</svg>

РЕЗУЛЬТАТ:

Прямоугольник

Задается тегом <rect>.

Атрибуты тега <rect>x Координата левой верхней точки прямоугольника по оси X

y Координата левой верхней точки прямоугольника по оси Y

width Ширина прямоугольника

height Высота прямоугольника

rx Радиус закругления углов прямоугольника по оси X

ry Радиус закругления углов прямоугольника по оси Y

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<rect x="5" y="5" width="500" height="50" rx="10" ry="20"

stroke="#b4241b" stroke-width="3" fill="#ffff00" />

</svg>

РЕЗУЛЬТАТ:

Круг

Задается тегом <circle>.

Атрибуты тега <circle>cx Координата центра круга по оси X

cy Координата центра круга по оси Y

r Радиус круга

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<circle cx="35" cy="35" r="30"

stroke="#b4241b" stroke-width="3" fill="#ffff00" />

</svg>

РЕЗУЛЬТАТ:

Эллипс

Задается тегом <ellipse>.

Атрибуты тега <ellipse>cx Координата центра эллипса по оси X

cy Координата центра эллипса по оси Y

rx Радиус эллипса по оси X

ry Радиус эллипса по оси Y

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="70">

<ellipse cx="80" cy="35" rx="70" ry="30"

stroke="#b4241b" stroke-width="3" fill="#ffff00" />

</svg>

РЕЗУЛЬТАТ:

Сложная траектория

Задается тегом <path>. Является самым универсальным из SVG-элементов. Позволяет создавать произвольные фигуры. Форма фигуры задается атрибутов d, значение которого – это набор специальных команд. Эти команды могут быть и в верхнем, и в нижнем регистре. Верхний регистр указывает на то, что применяется абсолютное позиционирование, а нижний – относительное.

Команды, определяющие траекторию и направление фигурной линииM, m Начальная точка

Mx,y

L, l Отрезок прямой

Lx,y

H, h Горизонтальная линия

Hx,y или hx

V, v Вертикальная линия

Vx,y или vy

A, a Дуга эллипса

Arx,ry x-axis-rotation large-arc-flag,sweep-flag x,y

rx,ry – радиусы дуги эллипса;

x-axis-rotation – угол поворота дуги относительно оси X;

large-arc-flag – если (=1), то строится большая части дуги, если (=0) – меньшая;

sweep-flag – если (=1), то дуга строится по часовой стрелке, если (=0) – против часовой стрелке;

x,y – координаты конечной точки дуги.

C, c Кубическая кривая Безье

Cx1,y1 x2,y2 x,y

x1,y1 – координаты первой контрольной точки;

x2,y2 – координаты второй контрольной точки;

x,y – координаты конечной точки кривой.

S, s Гладкая кубическая кривая Безье

Sx2,y2 x,y

x2,y2 – координаты второй контрольной точки;

x,y – координаты конечной точки кривой.

Первая контрольная точка является зеркальным отражением второй контрольной точки.

Q, q Квадратичная кривая Безье

Qx1,y1 x,y

x1,y1 – координаты контрольной точки;

x,y – координаты конечной точки кривой.

T, t Гладкая квадратичная кривая Безье

Qx1,y1 x,y

x,y – координаты конечной точки кривой.

Контрольная точка этой команды является зеркальным отражением контрольной точки предыдущей команды.

Z, z Замыкание траектории

Пример

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="600" height="100">

<path d="M10,15 h50 v60 L110,55 A25,35 -30 0,1 150,30

M160,50 C160,110 260,110 260,50 S360,-10 360,50

M370,50 Q420,100 470,50 T570,50 z"

stroke="#b4241b" stroke-width="3" fill="none" />

</svg>

РЕЗУЛЬТАТ:

Читать дальше: WML - язык разметки для мобильных телефонов

Популярное:

Генератор Sitemap

Анализ сайта

Чей IP-адрес?

Чей домен?

Где телефон?

Примеры Ajax

Примеры PHP

Примеры Javascript

Примеры HTML, CSS

GIF-аниматор

GEO-сервисы

Сайт в ТОП

Полезные ссылки

Генератор Robots.txt

Содержание:

HTML

Что такое HTML

<!DOCTYPE>

Мета теги в <head>

Тег <base>

Текст в html

HTML списки

Ссылки

Картинки на сайте

Таблицы

Фреймы

Формы

DHTML

Карты изображений

Графика в HTML

WML

SSI .shtml

Таблица цветов RGB

Правильное

сочетание цветов

Таблица

«безопасных»

цветов

Таблица символов

Примеры HTML, CSS

CSS

Таблицы стилей CSS

Свод стилей

Псевдоклассы CSS

Псевдоэлементы CSS

Адаптивная верстка

Flexbox

@media screen

JavaScript

PHP + MySQL

XML + XSLT

AJAX

Графика CorelDRAW

SEO

Сервисы

Разное

Шаблоны сайтов

Новое за неделю

Черный список IP

Вся история

заголовка документа.

HTML. Лекция 4. Графика. →

HTML. Лекция 3. Контейнеры тела документа

Опубликовано Февраль 16, 2012 автором IdeaFix

В этой лекции рассказывается об элементах разметки тела HTML-документа, подробно разбирается их типизация, назначение и применение.

Теги тела документа

Теги тела документа предназначены для управления отображением информации в программе интерфейса пользователя. Они описывают гипертекстовую структуру базы данных при помощи встроенных в текст контекстных гипертекстовых ссылок. Тело документа состоит из:

иерархических контейнеров и заставок;

заголовков (от Н1 до Н6);

блоков (параграфы, списки, формы, таблицы, картинки и т.п.);

горизонтальных отчеркиваний и адресов;

текста, разбитого на области действия стилей (подчеркивание, выделение, курсив);

математических описаний, графики и гипертекстовых ссылок.

Тело документа – контейнер ВОDY

Описание тегов тела документа следует начать с тега ВОDY. В отличие от тега НEАD, тег ВОDY имеет атрибуты.

Атрибут BАСКGROUND определяет фон, на котором отображается текст документа. Так, если источником для фона HTML- документа является графический файл image.gif, то в открывающем теге тела BODY появляется соответствующий атрибут:

<ВОDY ВАСКGROUND="image.gif">

Как видно из этого примера, в качестве значения данного атрибута используется адрес в сокращенной форме URL. В данном случае это адрес локального файла. Следует заметить, что разные интерфейсы пользователя поддерживают различные дополнительные атрибуты для тега ВОDY.

Атрибут Значение

ВGCOLOR=#FFFFFF Цвет фона

ТЕХТ=#0000FF Цвет текста

VLINK =#FF0000 Цвет пройденных гипертекстовых ссылок

LINK =#00FF00 Цвет гипертекстовой ссылки

В данной таблице строка #ХХХХХХ определяет цвет в терминах RGB в шестнадцатеричной нотации. Также имеется возможность задавать цвета по названию. Далее в таблице приведены названия цветов, определенные в стандарте HTML 4 и соответствующие им RGB-коды. Отметим, что многие современные браузеры выходят за рамки стандартов и поддерживают гораздо больше названий цветов.

Название Код Название Код

aqua #00FFFF navy #000080

black #000000 olive #808000

blue #0000FF purple #800080

fuchsia #FF00FF red #FF0000

gray #808080 silver #C0C0C0

green #008000 teal #008080

lime #00FF00 white #FFFFFF

maroon #800000 yellow #FFFF00

Так, значения атрибутов в таблице 3.1 определяют цвет текста как синий, фона — белый, пройденные ссылки красные, а новые ссылки зеленые. Если в качестве атрибутов тега ВОDY указать

<ВОDY ВGCOLOR=#FFFFFF ТЕХТ=#0000FF VLINK=#FF0000 LINK=#00FF00>,

то цвет фона будет белым, текст будет синим, ссылки — зелеными, а пройденные ссылки станут красными. Однако пользоваться этими атрибутами следует крайне осторожно, так как у пользователя может оказаться другой интерфейс, который эти параметры не интерпретирует.

Microsoft Internet Explorer и Netscape Navigator допускают применение атрибутов LEFTMARGIN=n и ТОРМАRGIN=n в теге <ВОDY>. Атрибут LEFTMARGIN= задает левое поле для всей страницы. ТОРМАRGIN= определяет верхнее поле. Число n показывает ширину поля в пикселах. Например, тег <ВОDY LEFTMARGIN =»40″> создаст на всей странице левое поле шириной 40 пикселов. При n, равном 0, левое поле отсутствует.

Теги управления разметкой

Заголовки

Заголовок обозначает начало раздела документа. В стандарте определено 6 уровней заголовков: от Н1 до Н6. Текст, окруженный тегами <Н1></Н1>, получается большим — это основной заголовок. Если текст окружен тегами <Н2></Н2>, то он выглядит несколько меньше (подзаголовок); текст внутри <НЗ></НЗ> еще меньше и так далее до <Н6></Н6>. Некоторые программы позволяют использовать большее число заголовков, однако реально более трех уровней встречается редко, а более 5 — крайне редко.

Ниже на рисунке показан результат использования следующих заголовков:

<H1>Заголовок 1</H1>

<H2>Заголовок 2</H2>

Тег <P>

Тег <P> применяется для разделения текста на параграфы. В нем используются те же атрибуты, что и в заголовках.

Атрибут АLIGN

Атрибут АLIGN позволяет выровнять текст по левому или правому краю, по центру или ширине. По умолчанию текст выравнивается по левому краю. Данный атрибут применим также к графике и таблицам.

Далее приведены возможные значения атрибута АLIGN:

АLIGN=justify выравнивание по левому и правому краям. Реализовано не во всех программах интерпретации.

АLIGN=left выравнивание по левому краю. По умолчанию текст HTML выравнивается по левому краю и не выравнивается по правому, то есть начало строк находится на одном уровне по вертикали, а концы — на разных. Чаще всего, получающийся при этом текст с равными промежутками между словами выглядит лучше. Поскольку выравнивание по левому краю задается автоматически, атрибут АLIGN=left можно опустить.

АLIGN=right выравнивание по правому краю. Текст, выравненный по правому краю и не выравненный по левому — концы строк находятся на одном уровне, а начало на разных, — часто используется с целью создать оригинальный дизайн. Для этого задается атрибут АLIGN=right в обычных тегах, например в теге <Р>.

АLIGN=center центрирование текста и графики. Есть несколько способов отцентрировать текст или графику. В спецификациях HTML 3.0 предлагается пользоваться тегом <АLIGN=сеntеr>. Однако этот тег применим не ко всем объектам HTML-страницы, поэтому разработчики Netscape добавили тег <СЕNТЕR>, который центрирует любые объекты и поддерживается браузерами Netscape Navigator 3.0, Microsoft Explorer 3.0 и другими. К тегу <СЕNТЕR> нужно относиться с осторожностью. Какой-нибудь браузер может его вообще проигнорировать, и на странице окажется текст, выравненный по левому краю.

Обтекание графики текстом. С помощью атрибута ALIGN можно заставить текст «обтекать» графический объект. Для этого следует поместить тег <IMG SRC=»/путь/файл.gif»> туда, где должен быть графический объект, и добавить атрибут ALIGN=left, ALIGN=right или АLIGN=center. Кроме того, с помощью атрибутов НSPAСЕ и VSPАСЕ (они описаны ниже) задается ширина горизонтальных и вертикальных полей, отделяющих изображение от текста. Можно также создать рамку вокруг картинки или обрамление таблицы текстом. Чтобы текст не «обтекал» графику, а прерывался, необходимо применить тег <BR> c атрибутом СLEAR.

Использование тега <ВR>

Принудительный перевод строки используется для того, чтобы нарушить стандартный порядок отображения текста. При обычном режиме интерпретации программа интерфейса пользователя отображает текст в рабочем окне, автоматически разбивая его на строки. В этом режиме концы строк текста игнорируются. Иногда для большей выразительности требуется начать печать с новой строки. Для этого и нужен тег ВR. Атрибут СLЕАR в теге <ВR> используется для того, чтобы остановить в указанной точке обтекание объекта текстом и затем продолжить текст в пустой области за объектом. Продолжающийся за объектом текст выравнивается в соответствии со значениями LEFT, RIGHT или АLL атрибута СLЕАR:

<BR СLЕАR=lеft> Текст будет продолжен, начиная

с ближайшего пустого левого поля.

<BR СLЕАR=right> Текст будет продолжен, начиная

с ближайшего пустого правого поля.

<BR СLЕАR=аll> Текст будет продолжен, как

только и левое, и правое поля окажутся пустыми.

Элемент разметки <NOВR>

Тег <NОВR> (Nо Вrеаk, без обрыва) дает браузеру команду отображать весь текст в одной строке, не обрывая ее. Если текст, заключенный в теги <NОВR>, не поместится на экране, браузер добавит в нижней части окна документа горизонтальную полосу прокрутки. Если вы хотите оборвать строку в определенном месте, поставьте там тег <ВR>.

Теги управления отображением символов

Все эти теги можно разбить на два класса: теги, управляющие формой отображения (font style), и теги, характеризующие тип информации (information type). Часто внешне разные теги при отображении дают одинаковый результат. Это зависит главным образом от настроек интерпретирующей программы и вкусов пользователя.

Теги, управляющие формой отображения

Курсив, усиление, подчеркивание, верхний индекс, нижний индекс, шрифт большой, маленький, красный, синий, различные комбинации — все это делает страницы более интересными. Microsoft Internet Explorer и Netscape Navigator позволяют определить шрифт с помощью тега FONT. Теперь можно объединять на одной странице несколько видов шрифтов, вне зависимости от того, какой из них задан по умолчанию в браузере пользователя.

Теги <ВIG> и <SMALL> — изменение размеров шрифта

Текст, расположенный между тегами <ВIG></ВIG> или <SMALL> </SMALL>, будет, соответственно, больше или меньше стандартного.

Верхние и нижние индексы

С помощью тегов <SUР> и <SUВ> можно задавать верхние и нижние индексы, необходимые для записи торговых знаков, символов копирайта, ссылок и сносок. Рассматриваемые теги позволяют создать внутри текстовой области верхние или нижние индексы любого размера. Чтобы они казались меньше окружающего текста, можно использовать теги <SUР> и <SUВ> с атрибутом FONT SIZE=-1, уменьшающим размер шрифта.

Атрибут SIZЕ

Атрибут SIZЕ тега <FОNТ> позволяет задавать размер текста в данной области. Если вы не пользуетесь тегом <BASEFONT SIZE=n> для задания определенного размера шрифта на всей странице, то по умолчанию принимается 3. Некоторые браузеры тег <FONТ> не поддерживают, поэтому желательно употреблять его только внутри текстовой области. В других случаях лучше использовать теги <Н1>, <Н2>, <НЗ> и т.д. Главное преимущество тега <FONТ> состоит в том, что после окончания действия он не разбивает строку, как теги <Нn>. Поэтому тег <FONТ> бывает очень полезен для изменения размера шрифта в середине строки.

Атрибут СОLОR

Если вы хотите сделать свою страницу более красочной, можете воспользоваться атрибутом СОLОR в теге FONТ, и тогда единственным ограничением будет цветовая палитра на компьютере пользователя.

Теги, управляющие формой отображения, приведены в таблице.

Тег Значение

<I>…</I> Курсив (Italic)

<B>…</B> Усиление (Вold)

<TT>…</TT> Телетайп

<U>…</U> Подчеркивание

<S>…</S> Перечеркнутый текст

<BIG>…</BIG> Увеличенный размер шрифта

<SMALL>…</SMALL> Уменьшенный размер шрифта

<SUB>…</SUB> Подстрочные символы

<SUP>…</SUР> Надстрочные символы

<ЕМ>…</ЕМ> Типографское усиление

<СIТЕ>…</СIТЕ> Цитирование

<STRONG>…</STRONG> Усиление

<СODE>…</СODE> Отображает примеры кода (например, "коды программ")

<SАМР>…</SАМР> Последовательность литералов

<КВD>…</КВD> Пример ввода символов с клавиатуры

<VAR>…</VAR> Переменная

<DFN>…</DFN> Определение

<Q>…</Q> Текст, заключенный в двойные кавычки

Эти теги допускают вложенность, поэтому все они имеют тег начала и конца. При использовании таких тегов следует помнить, что их отображение зависит от настроек программы-интерфейса пользователя, которые могут и не совпадать с настройками программы-разработчика гипертекста.

Создание списков в HTML

Списки являются важным средством структурирования текста и применяются во всех языках разметки. В НТМL имеются следующие виды списков: ненумерованный список (неупорядоченный) (Unordered Lists <UL>), нумерованный список (упорядоченный) (Ordered Lists <OL>) и список определений. Теги для ненумерованных и нумерованных списков — это основа HTML. HTML 3.2 добавляет несколько атрибутов к тегам списков для выбора разных типов маркеров в ненумерованных списках и разных схем нумерации в нумерованных. Можно включать такие атрибуты и в сами теги элементов списка (List Item <LI>), чтобы сменить тип маркера в середине списка. После появления нового атрибута все последующие маркеры в списке будут иметь такой же вид.

Неупорядоченные списки — тег <UL>

Ненумерованный список. Ненумерованный список предназначен для создания текста типа:

первый элемент списка;

второй элемент списка;

третий элемент списка.

Записывается данный список в виде последовательности:

<UL>

<LI>первый элемент списка

<LI>второй элемент списка

<LI>третий элемент списка

</UL>

Теги <UL> и </UL> — это теги начала и конца ненумерованного списка, тег <LI> (List Item) задает тег элемента списка. Помимо этих тегов, существует тег, позволяющий именовать списки — <LН> (List Header).

Атрибуты маркеров в ненумерованном списке

Чтобы не применять одни и те же маркеры на разных уровнях вложенности, можно использовать атрибут ТYРЕ. Вы можете задать любой тип маркера в произвольном месте списка. Можно даже смешивать разные типы маркеров в одном списке. Ниже перечислены теги с атрибутами стандартных маркеров:

<UL TYPE=DISK>Тег создает сплошные маркеры

такого типа, как в списках первого уровня по

умолчанию.

<UL TYPE=СIRCLE>Тег создает маркеры в виде

окружностей.

<UL TYPE=SQUARE>Тег создает сплошные квадратные

маркеры.

Упорядоченные списки — тег <OL>

Нумерованные списки. Тег <OL> вместе с атрибутом ТYРЕ= в HTML 3.2 позволяет создавать нумерованные списки, используя в качестве номеров не только обычные числа, но и строчные и прописные буквы, а также строчные и прописные римские цифры. При необходимости можно даже смешивать эти типы нумерации в одном списке:

<ОL ТYРЕ=l> Тег создает список с нумерацией

в формате 1., 2., 3., 4. и т.д.

<ОL ТYРЕ=А> Тег создает список с нумерацией

в формате А., В., С., D. и т.д.

<OL ТYРЕ=а> Тег создает список с нумерацией

в формате а., b., с., d. и т.д.

<ОL ТYРЕ=I> Тег создает список с нумерацией

в формате I., II., III., IV. и т.д.

Список определений — тег <DL>

Теги списка (Definition List: <DL>, <DT>, <DD>) используют для создания списка терминов и их определений. Схема использования тега следующая.

<DL><DT>Термин</DT> <DD>Определение</DD></DL>

Определяемый термин записывается на одной строке, а его определение — на следующей, с небольшим отступом вправо. Тег <DL> позволяет создавать отдельные абзацы с отступом без нумерации или маркеров. Отступ делается от левого края. Если на странице несколько тегов <DL>, то текст постепенно сдвигается все больше вправо. В конце определения поместите закрывающий тег </DL>. Помните, что тег lt;DL> сдвигает только левую границу абзаца.

Горизонтальные линейки — тег <НR>

Горизонтальное отчеркивание (Horizontal Rule) применяется для разделения документа на части. С помощью одного лишь тега <НR> можно придать странице оригинальный вид. Попробуйте поэкспериментировать с тегом <НR>, и вы получите линии, совсем не похожие на те, которыми обычно пользуетесь.

Преформатированный вывод — тег <РRЕ>

Применение этого тега позволяет отобразить текст «как есть» (без форматирования), теми же символами и с тем же разбиением на строки.

Применение тега <BLINK>

Текст, помещенный между тегами <BLINK> и </BLINK>, мерцает. Данный тег поддерживается только браузером Netscape Navigator. Пользоваться им следует с большой осторожностью.

Комментарии в языке HTML

При разметке документов HTML возникает необходимость в использовании комментариев, которые браузер не выводит на экран, но другой специалист, редактирующий данный документ, может прочитать. В таких примечаниях можно найти информацию о том, кто является автором документа, где и почему используется конкретный элемент HTML и т.п. Комментарии HTML начинаются с символа «<!—» и оканчиваются символом «—>». Можно вставлять текст с любыми символами. Комментарии могут состоять из нескольких строк текста. В общем и целом они ничем не отличаются от аналогичных комментариев в других языках программирования, так как видимы только тогда, когда это необходимо. Например, браузер игнорирует их. При создании файла HTML можно разместить в нем комментарии о его структуре. Кроме того, там можно размещать информацию о том, какие сложные операции способен выполнять данный документ.

Гипертекстовые ссылки

Все рассмотренные выше средства управления отображением текста, безусловно, важны, но они только дополняют основной тег HTML-документа — гипертекстовую ссылку. Для записи гипертекстовой ссылки используется тег <А>, который называют «якорь» (аnchor). Якорь имеет несколько атрибутов, главным из которых является НREF. Простую ссылку можно записать в виде

<А НREF="http://www.uspy.ru/">Отображаемое название гипертекстовой ссылки</А>,

где значение атрибута HREF — адрес документа «index.htm» на машине «www.intuit.ru», доступ к которой осуществляется по протоколу НТТР. Форма записи этого адреса называется универсальным локатором ресурсов URL и является составной частью технологии WWW.

Согласно схеме HTTP нотации URI, полный адрес информационного ресурса, доступного по протоколу HTTP, надлежит записывать следующим образом:

http://user:password@domain.ru:port/path/some.html?query\_string,

где http — протокол обмена данными; user — идентификатор пользователя; password – пароль; domain.ru — доменное имя сервера; port — номер TCP-порта, на котором ведет обслуживание сервер; path — путь в корневом каталоге сервера к файлу ресурса; some.html — файл ресурса;query\_string — поисковое предписание.

Заданный в таком виде адрес ресурса называется абсолютным или полным адресом ресурса. На практике редко используют все компоненты полного адреса схемы HTTP. Чаще всего первые компоненты опускают. Например, обращение к документу в том же каталоге в гипертекстовой ссылке будет записано просто как имя данного файла. Обращение к CGI-скрипту может выглядеть следующим образом:

<A HREF=../scripts/my\_script?query\_string>

Имя протокола, имя домена, номер порта и другие компоненты начала URL опущены. В этом случае говорят, что ссылка задана частично определенной или неполной формой URL.

Естественно, что браузер при обращении к серверу будет восстанавливать полную форму URL, опираясь на некоторую схему по умолчанию, которая называется базовым URL. Иногда неполную форму URL называют относительным URL, подразумевая, что адрес задается относительно некоторого базового адреса.

По умолчанию в качестве базового используется URL каталога, в котором находится текущий документ. Если URL начинается с символа «.» или «..», то это означает исчисление от текущего каталога. Если URL начинается с символа «/», то относительный URL берется от корня каталогов сервера.

В HTML есть элемент разметки BASE (рассмотренный ранее), который позволяет задать или переопределить базовый адрес. Первоначально этот контейнер использовался только в заголовке HTML-документа. Сейчас его применяют как за пределами документа (например, при создании документов HTML-редакторами), так и в теле документа.

Содержание контейнера гипертекстовой ссылки, заключенное между тегом начала и тегом конца, выделяется в тексте цветом, определенным для контекстных гипертекстовых ссылок. В атрибутах тега <ВОDY>:

Атрибут Значение

ТЕХТ=#000000 Цвет текста (черный)

ALINK=#FF0000 Цвет «активных» гипертекстовых ссылок (красный)

VLINK=#FF00FF Цвет пройденных гипертекстовых ссылок (пурпурный)

LINK=#0000FF Цвет гипертекстовой ссылки (синий)

Одна из особенностей создания Web-сайта состоит в том, что представленную на нем информацию желательно разбить на отдельные части, которые могут быть выведены на экран без необходимости его прокрутки. Организация связей между отдельными частями осуществляется с помощью гипертекстовых ссылок.

<A HREF="http://www.uspu.ru/students">Студенту</A>

При нажатии на ссылку в окно браузера будет загружен новый документ.

Другой формой использования тега <А> является определение точек внутри текста, на которые можно сослаться. Такой метод применяется в том случае, когда документ нельзя поделить на части и необходимо быстро перемещаться из оглавления в текст:

<А NАМЕ="роint">

Для ссылки на такую точку используют следующую форму URL:

<А НREF="http://www.uspu.ru/index.html#роint">Ссылка на точку "роint" в документе "index.html"</А>

На описании простых гипертекстовых ссылок обзор средств языка НТМL, ориентированных на текстовое представление информации и организацию гипертекстовых баз данных, можно закончить.

Запись опубликована в рубрике HTML с метками html, WEB, Лекция. Добавьте в закладки постоянную ссылку.

← HTML. Лекция 2. Структура HTML-документа и элементы разметки заголовка документа.

HTML. Лекция 4. Графика.

Опубликовано Февраль 16, 2012 автором IdeaFix

В этой лекции рассматриваются принципы применения графических образов при HTML-разметке.

Использование графики в HTML

Для того чтобы вставить в Web-страницу изображение, необходимо либо нарисовать его, либо взять уже готовое. В любой программе рисования можно создать простое изображение и сохранить его в нужном формате. Если программа этот формат не поддерживает, необходимо преобразовать файл в требуемый формат. Существует множество программ, предназначенных для преобразования одного графического формата в другой. Позаимствовать же картинки можно из различных программных пакетов или с других Web-страниц в Internet, содержащих библиотеки свободного доступа художественных изображений. Когда браузер выводит Web-страницу с изображением, соответствующий графический файл временно хранится в памяти компьютера. В большинстве браузеров есть команда, позволяющая сохранить файл на локальном диске. Существует также множество других вариантов получения графических файлов.

Изображения могут нести определенную информацию, да и просто придают Web-странице привлекательный вид. Приведем наиболее распространенные случаи применения изображений:

логотип компании на деловой странице;

графика для рекламного объявления;

различные рисунки;

диаграммы и графики;

художественные шрифты;

подпись автора страницы;

применение графической строки в качестве горизонтальной разделительной линии;

применение графических маркеров для создания красивых маркированных списков.

Теперь рассмотрим как вставить изображение в Web-страницу. Тегом HTML, который заставляет браузер выводить изображение, является <IMG> с обязательным атрибутом SRC (SouRCe, источник). Имя файла представляет собой имя выводимого графического файла. Замыкающего тега не требуется.

Пример вставки изображения:

<IMG SRC="image.gif" ALT="ИЗОБРАЖЕНИЯ">

Изображения на Web-странице могут использоваться в качестве гипертекстовых ссылок, как и обычный текст. Читатель щелкает на изображении и отправляется на другую страницу или переходит к другому изображению. Для обозначения изображения как гипертекстовой метки используется тот же тег <A>, что и для текста, но между <A> и </A> вставляется тег изображения <IMG> :

<A HREF="адрес файла или изображения"> <IMG SRC="image.gif"></A>

При этом изображение, используемое в качестве гипертекстовой ссылки, обводится дополнительной рамкой.

Атрибуты и их аргументы. Тег изображения имеет один обязательный атрибут SRC и необязательные: ALT, ALIGN, USEMAP, HSPACE, VSPACE, BORDER, WIDTH, HEIGHT.

Атрибут SRC

Указывает файл изображения и путь к нему; изображение должно быть загружено в браузер и размещено в том месте документа, где расположен тег изображения.

Атрибут ALT

Позволяет указать текст, который будет выводиться вместо изображения браузерами, неспособными представлять графику. В некоторых случаях при недостаточной пропускной способности линий связи пользователи отключают отображение графики. Наличие названий вместо картинок облегчает восприятие Web-страниц в таком режиме.

Атрибут АLIGN

Определяет положение изображения относительно окружающего его текста. Возможные значения аргумента — [«top» | «middle» | «bottom»] (соответственно, «вверху», «посередине», «внизу»).

ALIGN=»top» выравнивает верх изображения по верхнему краю самого высокого элемента в строке окружающего текста.

ALIGN=»middle» выравнивает центр изображения по базовой линии строки окружающего текста.

ALIGN=»bottom» выравнивает нижний край изображения по базовой линии строки окружающего текста.

Кроме основных значений атрибута ALIGN=»ключевое слово» существует еще ряд аргументов, которые расширяют возможности взаимного размещения графики и текста. Рассмотрим их подробнее.

Дополнительные возможные значения аргумента — [«left» | «right» | «top» | «texttop» | «middle» |»absmiddle» | «baseline» | «bottom» | «absbottom» ].

ALIGN=»left» определяет огибаемое текстом изображение. Изображение располагается вдоль левой границы документа, а последующие строки текста огибают его справа.

ALIGN=»right» определяет огибаемое текстом изображение. Изображение располагается вдоль правой границы документа, а последующие строки текста огибают его слева.

ALIGN=»top» выравнивает верх изображения по верхнему краю самого высокого элемента в строке окружающего текста точно так же, как при использовании стандартного набора атрибутов.

ALIGN=»texttop» выравнивает верх изображения по верхнему краю самого высокого текстового символа в строке окружающего текста. Действие этого аргумента в большинстве случаев, но не всегда, подобно действию аргумента ALIGN=»top».

ALIGN=»middle» выравнивает центр изображения по базовой линии строки окружающего текста точно так же, как при использовании стандартного набора атрибутов.

ALIGN=»absmiddle» выравнивает центр изображения по центру строки окружающего текста.

ALIGN=»baseline» выравнивает нижний край изображения по базовой линии строки окружающего текста, то есть производит такое же действие, как и ALIGN=»bottom».

ALIGN=»bottom» выравнивает нижний край изображения по базовой линии строки окружающего текста точно так же, как при использовании стандартного набора атрибутов.

ALIGN=»absbottom» выравнивает нижний край изображения по нижнему краю строки окружающего текста.

Атрибут USEMAP

Если присутствуют атрибут USEMAP и тег <MAP>, изображение становится чувствительной картой, или «графическим меню». Если щелкнуть кнопкой мыши на активной области изображения, для которого определен атрибут USEMAP, произойдет гипертекстовый переход к информационному ресурсу, установленному для этой области. Более подробно этот вопрос будет рассматриваться в следующем разделе.

Атрибут BORDER

Целочисленное значение аргумента определяет толщину рамки вокруг изображения. Если значение равно нулю, рамка отсутствует. Чтобы не вводить пользователей в заблуждение, не стоит задействовать BORDER=0 в изображениях, которые представляют собой часть элемента якоря, поскольку рисунки, применяемые в качестве гиперссылок, обычно выделяются цветной рамкой.

Атрибут HSPACE

Целочисленное значение этого атрибута задает горизонтальное расстояние между вертикальной границей страницы и изображением, а также между изображением и огибающим его текстом.

Атрибут VSPACE

Целочисленное значение этого атрибута задает вертикальное расстояние между строками текста и изображением.

Атрибуты WIDTH и HEIGHT

Оба атрибута задают целочисленные значения размеров изображения по горизонтали и по вертикали соответственно. Это позволяет уменьшить время загрузки страницы с графикой. Браузер сразу отводит рамку для изображения и продолжает загружать текст на страницу. Пока загружается графика, пользователь может начать читать текст. Определить размер изображения нетрудно, для этого достаточно воспользоваться любой программой просмотра графических файлов, например ACDSee или графическим редактором Corel PhotoPaint или Adobe Photoshop. Откройте файл в графическом редакторе и определите размер картинки в пикселах. В теге изображения следует указать ширину и высоту картинки.

<IMG SRC="image.gif" ALT="изображение" WIDTH="100" HEIGHT="200" HSPACE="10" VSPACE="10"

BORDER="2" ALIGN="left">

Форматы графических файлов

Самыми распространенными графическими форматами в Web являются GIF и JPEG. GIF — наиболее подходящий формат для обмена изображениями между системами. Архивы с изображениями в формате GIF можно найти на многих серверах Internet. Данный формат поддерживают многие графические приложения, в том числе все программы просмотра графики World Wide Web.

Однако у этого формата есть одно серьезное ограничение: он не поддерживает изображения с глубиной цвета больше восьми бит на пиксел. Обычно восьми бит на пиксел оказывается достаточно для контурных изображений типа комиксов и рисунков, где используется ограниченное количество цветов, или для небольших картинок, где для цветопередачи хватает 256 оттенков. Однако для больших изображений фотографического качества больше подходит формат JPEG.

Формат GIF использует один из лучших алгоритмов сжатия LZW, который изначально не предназначался специально для графики. Он не очень подходит для работы с двухцветными (черно-белыми) или фотографическими изображениями.

С развитием аппаратного обеспечения, поддерживающего высокое разрешение и богатую цветовую гамму, графические файлы значительно увеличились в размерах. Профессиональные художники теперь, как правило, работают с файлами, содержащими 10 и более мегабайт данных на каждое изображение. Даже пользователи с более скромными запросами подчас имеют дело с изображениями 640 на 480 пикселов в 256 цветах (а это более 300 килобайт). Кроме того, многие сейчас начинают работать с полноцветными изображениями 1024 на 768 пикселов (это более 2,3 мегабайт данных). Так как высококачественные изображения встречаются все чаще, ограничения, накладываемые традиционными методами сжатия (например, LZW), становятся все более ощутимыми.

Для поиска оптимального способа сжатия изображений фотографического качества две международные организации по стандартизации, International Telecommunications Union (ITU, Международный союз телекоммуникаций) и International Organization for Standartization (ISO, Международная организация по стандартизации), создали Joint Photographic Experts Group (JPEG, объединенная экспертная группа по фотографии). С тех пор сокращение «JPEG» используется как название этой техники сжатия. Кроме того, оно входит в названия некоторых использующих ее файловых форматов.

Имя JPEG указывает на метод сжатия, но не на формат файла. На самом деле метод сжатия JPEG используют как многочисленные мало различающиеся форматы, зачастую известные, например «JPEG», так и единичные радикально отличающиеся форматы, такие как TIFF и Quick Time. К счастью, все же большинство форматов, известных под именем «JPEG», очень похожи, и, скорее всего, у вас не возникнет с этим проблем, однако знать о возможных осложнениях не помешает.

Формат JPEG отличается от других графических форматов прежде всего тем, что он использует метод сжатия «с потерями». JPEG частично идентифицирует и удаляет ту информацию, которая несущественна для восприятия изображения. В результате JPEG может достигать высокого уровня сжатия без заметных потерь в качестве изображения.

Метод сжатия «с потерями» имеет много реализаций. JPEG достигает существенного сжатия за счет отбрасывания той графической информации, которая обычно не проявляется в реальных изображениях. Однако при сжатии с помощью JPEG изображений с четкими контурами линии начинают заметно «дрожать». Так, например, если изображение содержит какие-либо подписи, подобный эффект может возникнуть вокруг символов. Этот эффект можно свести к минимуму, задав очень высокие значения параметра качества, однако при этом нельзя достичь приемлемых показателей сжатия.

Так как JPEG предполагает сжатие с потерями, при создании файлов необходимо быть внимательным. Большинство программ, создающих такие файлы, позволяют задавать значение параметра качества изображения. Обычно оно варьирует от нуля до ста. Нижние значения позволяют при сжатии JPEG отбрасывать больше информации, в результате чего получаются файлы меньшего размера. В свою очередь, высокие значения ограничивают количество информации, которой можно пренебречь во время сжатия.

Одна из наиболее распространенных ошибок заключается в интерпретации значения параметра качества от нуля до ста как процента сохраняемых данных. Чтобы развеять это заблуждение, некоторые современные программные продукты JPEG просто используют шкалу «лучшее сжатие» — «лучшее качество».

Хитрость заключается в том, чтобы при наименьшей величине параметра качества получить изображение без видимого его ухудшения. Лучше начинать со средних значений и внимательно оценивать результат. Если вы отмечаете некоторое ухудшение, попробуйте увеличить значение параметра, если нет — попытайтесь его уменьшить. При просмотре изображения обращайте внимание на следующие моменты: четкость очертаний и углов, например вокруг текста, или контур детали изображения, выделяющейся на общем фоне. Часто бывает заметно, что контур «смазан» или линия «дрожит».

Сжатие JPEG использует мозаику размером восемь на восемь пикселов. Если задаются слишком низкие значения качества, ее границы могут стать заметны. Если у вас уже есть изображения в GIF или другом восьмиразрядном формате, возможно, вы захотите попробовать конвертировать их в JPEG. Несмотря на то, что иногда это все же приводит к уменьшению необходимого для хранения файлов пространства, в большинстве случаев игра не стоит свеч. Если вы все же хотите попытаться, сначала выясните, сколько цветов использует изображение GIF. Если в нем только 64 цвета, то конверсия вряд ли себя оправдает, так как изображение с такой бедной цветовой палитрой не имеет тех плавных цветовых переходов, которые хорошо сжимает JPEG. В результате вы просто ухудшите качество изображения, не освободив места.

Одна из серьезных проблем конверсии изображений GIF в JPEG заключается в том, что изображения в формате GIF, лимитированные набором из 256 (или менее) цветов, часто используют клиширование (dithering) и полутона (halftoning), в результате чего пикселы двух цветов смешиваются для получения эффекта третьего тона. В результате использования этой техники образуются шаблоны, крайне плохо сжимаемые с помощью JPEG. Отдельные программы позволяют усреднять значения этих шаблонов, «смягчая» таким образом изображение до преобразования, в результате чего сжатие с помощью JPEG оказывается более эффективным.

Активные изображения

Активные изображения (image maps), или изображения, чувствительные к щелчкам мыши, позволяют создать на узле графические меню произвольной формы. Активное изображение — это изображение с так называемыми активными областями (hot spots), которые ссылаются на URL других страниц или узлов.

Есть два метода формирования активных изображений: на сервере и у клиента. Изображения первого типа используют сервер для того, чтобы найти соответствующий данной активной области URL и передать на браузер нужную страницу. Активные изображения, работающие на клиентской машине, задают информацию об активной области на НТМL-странице, так что браузер сам выясняет, какие области являются активными, и запрашивает с сервера соответствующую страницу.

Активные изображения, работающие у клиента, имеют несколько преимуществ. Во-первых, страницы с ними можно перенести на другой сервер. Во-вторых, серверу не приходится выполнять лишнюю работу (например, просматривать всю информацию об активных областях), то есть нагрузка на сервер уменьшается. При использовании работающих на сервере активных изображений в каталоге сgi-bin сервера должен быть соответствующий сценарий. Из соображений безопасности многие системные администраторы не записывают сценарии в каталог сgi-bin. Поэтому более подробно мы рассмотрим создание активных изображений у клиента.

Создание активного изображения. Процесс создания активного изображения состоит из двух этапов. Сначала необходимо определить на картинке области, которые нужно сделать активными, а потом соотнести их со ссылками на другие URL. Активные области задаются перечислением их координат (в пикселах). Все это можно сделать вручную, определив координаты углов активных областей, но гораздо проще воспользоваться какой-нибудь программой, например МарЕdit.

Определить карту легко. Нужно открыть в МарEdit HTML-файл, содержащий изображение, на котором требуется создать активные области, после чего выбранное изображение будет загружено в рабочее окно. Затем следует выбрать тип активной области (квадрат, треугольник и круг), щелкнуть и потянуть мышкой, обозначив границу области. Программа автоматически производит запись в HTML-файл, описывающий границы активной области. Затем этой области нужно приписать URL. В любых местах изображения можно нарисовать активные области и определить для каждой из них URL. Важно оставлять между областями немного места, чтобы при чтении быть уверенным, что активизируется правильная ссылка. Границы активных областей задаются координатами углов прямоугольника и многоугольника или центра и радиуса круга. Если вы решили делать активное изображение у клиента, Map Edit поставляет данные только для тегов <MAP>. Вам придется самим задать тег изображения с атрибутом USEMAP и поместить его после тега

</MAP>. Не забудьте перед именем карты в

атрибуте USEMAP записать символ «#» следующим образом:

<IMG SRC="mymap.gif" USEMAP="#sitemap">

Активные изображения у клиента работают независимо от программного обеспечения сервера и не перестанут функционировать, даже если файлы будут перенесены на другой сервер. Таким изображениям требуются только две вещи: браузер, поддерживающий HTML 3.0, и информация о карте, записанная в HTML-файле. Приведем пример активных изображений.

<IMG SRC="image.gif" ALT="Изображения" USEMAP="#imap">

<MAP NAME="imap">

<AREA SHAPE="rect" COORDS="0,0,100,100" HREF="http://www.intuit.ru/help/index.html">

<AREA SHAPE="rect" COORDS="100,0,200,100" HREF="http://www.intuit.ru/shop/index.html">

<AREA SHAPE="default" nohref>

</MAP>

Изображения в миниатюре

Часто для иллюстрации какой-то темы требуются изображения большого размера, загружаться они будут достаточно долго. В том месте, где требуется разместить большой рисунок, можно поместить маленькую его копию и сделать ссылку на полномасштабное изображение. Те посетители, которым это действительно интересно, смогут посмотреть изображение полностью, а все остальные пролистнут страницу, не задерживаясь. Такая методика особенно хороша для обложек книг, фотографий, рекламных листков, которые не все читатели захотят изучить в деталях.

Запись опубликована в рубрике HTML с метками html, WEB, Лекция. Добавьте в закладки постоянную ссылку.

← HTML. Лекция 3. Контейнеры тела документа

ideafix.name

Перейти к содержимому

Главная

← HTML. Лекция 4. Графика.

HTML. Лекция 6. HTML-формы. →

HTML. Лекция 5. Таблицы.

Опубликовано Февраль 16, 2012 автором IdeaFix

В этой лекции подробно рассматриваются принципы применения таблиц в HTML-разметке. Это и табличная организация текста, и табличная координатная сетка, и организованная в таблицы графика.

Средства описания таблиц в HTML

По мере развития WWW стало ясно, что средств, которые заложены в НТМL, недостаточно для качественного отображения различного типа документов. Недостатком НТМL было отсутствие в его составе средств отображения таблиц. Для этой цели обычно использовался предформатированный текст (тег <PRE>), в котором таблица обрисовывалась символами АSСII. Но такая форма представления таблиц была недостаточно высокого качества и выбивалась из общего стиля документа. После введения таблиц в HTML у Web-мастеров появился не просто инструмент для размещения текстовых и числовых данных, а мощное средство дизайна для размещения в нужном месте экрана графических образов и текста.

Создание таблиц в HTML

Для описания таблиц используется тег <ТАВLЕ>. Тег <ТАВLЕ>, как и многие другие, автоматически переводит строку до и после таблицы.

Создание строки таблицы — тег <ТR>

Тег <ТR> (Таble Row, строка таблицы) создает строку таблицы. Весь текст, другие теги и атрибуты, которые требуется поместить в одну строку, должны размещаться между тегами lt;ТR></ТR>.

Определение ячеек таблицы — тег <ТD>

Внутри строки таблицы обычно размещаются ячейки с данными. Каждая ячейка, содержащая текст или изображение, должна быть окружена тегами <ТD></ТD>. Число тегов <ТD></ТD> в строке определяет число ячеек

<HTML>

<BODY>

<H1 ALIGN=center>Таблица</H1>

<CENTER>

<TABLE BORDER>

<TR>

<TD COLSPAN=3>Если в таблице два

тега TR, то в ней две строки.</TD>

</TR>

<TR>

<TD>Если в строке три тега TD,</TD>

<TD>то в ней</TD>

<TD>три столбца.</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Использование таблиц в дизайне

Заголовки столбцов таблицы — тег <ТН>

Заголовки для столбцов и строк таблицы задаются с помощью тега заголовка <ТН></ТН> (Таblе Неаder, заголовок таблицы). Эти теги подобны <ТD></ТD>. Отличие состоит в том, что текст, заключенный между тегами <ТН></ТН>, автоматически записывается жирным шрифтом и по умолчанию располагается посередине ячейки. Центрирование можно отменить и выровнять текст по левому или правому краю. Если воспользоваться <ТD></ТD> с тегом <В> и атрибутом <АLIGN=center>, текст тоже будет выглядеть как заголовок. Однако следует иметь в виду, что не все браузеры поддерживают в таблицах жирный шрифт, поэтому лучше задавать заголовки таблиц с помощью <ТН>.

<HTML>

<BODY>

<TABLE BORDER>

<TR>

<TH>Заголовок центрирован по умолчанию

</TH>

<TH COLSPAN=2>Заголовок может объединять

столбцы</TH>

</TR>

<TR>

<TH>Заголовок может быть расположен

перед столбцами</TH>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TH ROWSPAN=3>Заголовок может объединять

строки</TH>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

</BODY>

</HTML>

Использование заголовков таблицы — тег <САРТIОN>

Тег <CAPTION> позволяет создавать заголовки таблицы. По умолчанию заголовки центрируются и размещаются либо над (<САРТION АLIGN=top>), либо под таблицей (<САРТION ALIGN=bottom>). Заголовок может состоять из любого текста и изображений. Текст будет разбит на строки, соответствующие ширине таблицы. Иногда тег <САРТION> используется для подписи под рисунком. Для этого достаточно описать таблицу без границ.

<HTML>

<BODY>

<TABLE BORDER>

<CAPTION ALIGN=top>Заголовок над таблицей

</CAPTION>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

<TABLE BORDER>

<CAPTION ALIGN=bottom>Заголовок под таблицей

</CAPTION>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

</BODY>

</HTML>

Атрибут NOWRAP

Обычно любой текст, не помещающийся в одну строку ячейки таблицы, переходит на следующую строку. Однако при использовании атрибута NOWRAP с тегами <ТН> или <ТD> длина ячейки расширяется настолько, чтобы заключенный в ней текст поместился в одну строку.

Атрибут СОLSPAN

Теги <ТD> и <ТН> модифицируются с помощью атрибута СОLSPAN (Column Span, соединение столбцов). Если вы хотите сделать какую-нибудь ячейку шире, чем верхняя или нижняя, можно воспользоваться атрибутом СОLSPAN, чтобы растянуть ее над любым количеством обычных ячеек.

<HTML>

<BODY>

<CENTER>

<TABLE BORDER=3>

<TR>

<TD>Если вы хотите сделать какую-нибудь

ячейку шире, чем верхняя или нижняя,

</TD>

<TD>можно воспользоваться атрибутом

СОLSPAN=2, </TD>

</TR>

<TR>

<TD BGCOLOR=white COLSPAN=2>чтобы

растянуть ее над любым количеством

обычных ячеек.</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Атрибут ROWSPAN

Атрибут ROWSPAN, используемый в тегах <ТD> и <ТН>, подобен атрибуту СОLSPAN=, только он задает число строк, на которые растягивается ячейка. Если вы указали в атрибуте ROWSPAN=s число, большее единицы, то соответствующее количество строк должно находиться под растягиваемой ячейкой. Внизу таблицы ее поместить нельзя.

Атрибут WIDТН

Атрибут WIDТН применяется в двух случаях. Можно поместить его в тег <ТАВLЕ>, чтобы дать ширину всей таблицы, а можно использовать в тегах <ТD> или <ТН>, чтобы задать ширину ячейки или группы ячеек. Ширину можно указывать в пикселах или в процентах. Например, если вы задали в теге <ТАВLЕ> WIDTH=250, вы получите таблицу шириной 250 пикселов независимо от размера страницы на мониторе. При задании WIDТН=50% в теге <ТАВLЕ> таблица будет занимать половину ширины страницы при любом размере изображения на экране. Так что, указывая ширину таблицы в процентах, имейте в виду, что если у пользователя узкая область просмотра, ваша страница может выглядеть несколько странно. Если вы пользуетесь пикселами, и таблица оказывается шире области просмотра, внизу появится полоса прокрутки для перемещения вправо и влево по странице. В зависимости от поставленных задач и тот, и другой способ задания ширины таблицы может оказаться полезным.

<HTML>

<BODY>

<TABLE BORDER WIDTH=100%>

<TR>

<TD ALIGN=center>Текст или данные -

ширина 100%</TD>

</TR>

</TABLE>

или<BR>

<TABLE BORDER WIDTH=50%>

<TR>

<TD ALIGN=center>Текст или данные -

ширина 50%</TD>

</TR>

</TABLE>

или<BR>

<TABLE BORDER WIDTH=200>

<TR>

<TD ALIGN=center>Текст или данные -

ширина 200 пикселов</TD>

</TR>

</TABLE>

или<BR>

<TABLE BORDER WIDTH=100>

<TR>

<TD ALIGN=center>Текст или данные -

ширина 100 пикселов</TD>

</TR>

</TABLE>

</BODY>

</HTML>

Применение пустых ячеек

Если ячейка не содержит данных, она не будет иметь границ. Если требуется, чтобы у ячейки были границы, но не было содержимого, необходимо поместить в нее что-то, что не будет видно при просмотре. Можно воспользоваться пустой строкой <ВR>. Можно даже задать пустые столбцы, определив их ширину в пикселах или относительных единицах и не введя в полученные ячейки никаких данных. Это средство может оказаться полезным при размещении на странице текста и графики.

Атрибут СЕLLРАDDING

Данный атрибут определяет ширину пустого пространства между содержимым ячейки и ее границами, то есть задает поля внутри ячейки.

<HTML>

<BODY>

<CENTER>

<TABLE BORDER CELLPADDING=20>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

<BR>

<TABLE BORDER CELLPADDING=0>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Атрибуты АLIGN и VALIGN

Теги <ТR>, <ТD> и <ТН> можно модифицировать с помощью атрибутов ALIGN и VALIGN. Атрибут АLIGN определяет выравнивание текста и графики по горизонтали, то есть по левому или правому краю, либо по центру. Горизонтальное выравнивание может быть задано несколькими способами:

ALIGN=blееdleft прижимает содержимое ячейки вплотную к левому краю.

ALIGN=left выравнивает содержимое ячейки по левому краю с учетом отступа, заданного атрибутом СЕLLPADDING.

АLIGN=сеnter располагает содержимое ячейки по центру.

АLIGN=right выравнивает содержимое ячейки по правому краю с учетом отступа, заданного атрибутом СЕLLPADDING.

<HTML>

<BODY>

<TABLE BORDER WIDTH=100%>

<TR>

<TD ALIGN=left>Текст или данные</TD>

<TD ALIGN=center>Текст или данные</TD>

<TD ALIGN=right>Текст или данные</TD>

</TR>

<TR>

<TD ALIGN=right>Текст или данные</TD>

<TD ALIGN=center>Текст или данные</TD>

<TD ALIGN=left>Текст или данные</TD>

</TR>

<TR>

<TD ALIGN=right>Текст или данные</TD>

<TD ALIGN=right>Текст или данные</TD>

<TD ALIGN=right>Текст или данные</TD>

</TR>

<TR>

<TD ALIGN=center>Текст или данные</TD>

<TD ALIGN=center>Текст или данные</TD>

<TD ALIGN=center>Текст или данные</TD>

</TR>

<TR>

<TD ALIGN=left>Текст или данные</TD>

<TD ALIGN=left>Текст или данные</TD>

<TD ALIGN=left>Текст или данные</TD>

</TR>

</TABLE>

</BODY>

</HTML>

Атрибут VALIGN осуществляет выравнивание текста и графики внутри ячейки по вертикали. Вертикальное выравнивание может быть задано несколькими способами:

VALIGN=top выравнивает содержимое ячейки по ее верхней границе.

VALIGN=middle центрирует содержимое ячейки по вертикали.

VALIGN=bottom выравнивает содержимое ячейки по ее нижней границе.

<HTML>

<BODY>

<CENTER>

<TABLE BORDER WIDTH=90%>

<TR>

<TD WIDTH=100>Атрибут VALIGN осуществляет

выравнивание текста и графики внутри

ячейки по вертикали.</TD>

<TD VALIGN=top>верх,</TD>

<TD VALIGN=middle>середина,</TD>

<TD VALIGN=bottom>низ.</TD>

</TR>

<TR VALIGN=top>

<TD> VALIGN=top Выравнивает содержимое

ячейки по ее верхней границе.</TD>

<TD>верх,</TD>

<TD>верх,</TD>

<TD>верх.</TD>

</TR>

<TR VALIGN=middle>

<TD>VALIGN=middle Центрирует содержимое

ячейки по вертикали.</TD>

<TD>середина,</TD>

<TD>середина,</TD>

<TD>середина.</TD>

</TR>

<TR VALIGN=bottom>

<TD>VALIGN=bottom Выравнивает содержимое

ячейки по ее нижней границе.</TD>

<TD>низ,</TD>

<TD>низ,</TD>

<TD>низ.</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Атрибут BORDER

В теге <ТАВLЕ> часто определяют, как будут выглядеть рамки, то есть линии, окружающие ячейки таблицы и саму таблицу. Если вы не зададите рамку, то получите таблицу без линий, но пространство под них будет отведено. Того же результата можно добиться, задав <ТАВLЕ ВОRDER=0>. Иногда хочется сделать границу потолще, чтобы она лучше выделялась. Можно для привлечения внимания к рисунку или тексту задать исключительно жирные границы. При создании вложенных таблиц приходится делать для разных таблиц границы различной толщины, чтобы их легче было различать.

Атрибут CELLSPACING

Атрибут СЕLLSPACING определяет ширину промежутков между ячейками в пикселах. Если этот атрибут не указан, по умолчанию задается величина, равная двум пикселам. С помощью атрибута СЕLLSPACING= можно размещать текст и графику там, где вам нужно. Если вы хотите оставить пустое место, можно вписать в ячейку пробел.

<HTML>

<BODY>

<CENTER>

<TABLE BORDER CELLSPACING=20>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

<TABLE BORDER CELLSPACING=10>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

<TABLE BORDER CELLSPACING=0>

<TR>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

</TR>

<TR>

<TD>Текст или данные</TD>

<TD></TD>

<TD>Текст или данные</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Атрибут BGCOLOR

Данный атрибут позволяет установить цвет фона. В зависимости от того, с каким тегом (TABLE, TR, TD) он применяется, цвет фона может быть установлен для всей таблицы, для строки или для отдельной ячейки. Значением данного атрибута является RGB-код или стандартное название цвета.

<HTML>

<BODY>

<CENTER>

<TABLE BORDER BGCOLOR=yellow>

<TR BGCOLOR=blue>

<TD>Текст или данные</TD>

<TD BGCOLOR=red>Текст или данные

</TD>

<TD>Текст или данные</TD>

</TR>

<TR BGCOLOR=green>

<TD>Текст или данные</TD>

<TD>Текст или данные</TD>

<TD BGCOLOR=lime>Текст или данные

</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Атрибут BACKGROUND

Данный атрибут задает фоновое изображение для таблиц. Применим к тегам TABLE и TD. Его значением является URL файла с фоновым изображением. Применение этого атрибута рассматривается ниже.

Использование таблиц в дизайне страницы

Таблицы хороши тем, что при желании можно сделать их границы невидимыми. Это позволяет с помощью тега <ТАВLЕ> красиво размещать на странице текст и графику. Пока тег <ТАВLЕ> остается единственным мощным средством форматирования в HTML. Дизайнеры Web-страниц сейчас обладают практически той же свободой в отношении использования «пустого пространства», что и создатели печатных страниц. Таблицы лучше всего помогают отойти от иерархического размещения текста на Web-страницах.

Если браузер поддерживает таблицы, он обычно правильно отображает наиболее интересные эффекты, полученные с их помощью

<HTML>

<BODY>

<CENTER>

<TABLE CELLPADDING="10" CELLSPACING="0"

BORDER="16">

<TR>

<TD ALIGN="center">

<H2>Уральский государственный

педагогический университет</H2>

<H3>Добро пожаловать!</H3>

<TABLE BORDER WIDTH="100%">

<TR>

<TD ALIGN="center"><I>Учебный курс

"Основы WEB-мастеринга"</I></TD>

</TR>

</TABLE>

</TD>

</TR>

</TABLE>

</CENTER>

</BODY>

</HTML>

Использование таблиц в дизайне

Создание разноцветных таблиц

Некоторые браузеры позволяют отображать цвета. Есть несколько способов раскрасить таблицу, в основном они зависят от используемого браузера.

Цветные границы в Netscape Navigator. Вы не только можете окружить таблицу красивой рамкой, но еще и задать для нее цвет, отличный от цветов текста и фона. Создайте простой серый GIF (или любой GIF, который вы хотели бы иметь в качестве фона) и определите его в теге <ВODY> как фон страницы. Затем задайте цвет фона страницы. В результате ваш тег <ВОDY> будет выглядеть примерно так:

<BODY ВАСКGROUND="image.gif" ВGCOLOR="#FF0000">

Вы создали двойной фон — GIF и заданный цвет. В результате фоновый цвет будет виден на всех границах таблиц и горизонтальных линиях (<НR>). Вне зависимости от того, является ваш фоновый GIF серым или нет, цветные линии и границы таблиц будут заметно выделяться. Если фоновый GIF устроен не слишком сложно, время загрузки страницы возрастет лишь немного.