

## Урок 3

# Графика

Одна из наиболее привлекательных черт WWW — возможность включения ссылок на графические файлы в HTML-документ. Графика — лучшее украшение веб-узла. Под графикой подразумевают значки, рисунки, фотографии и карты изображений, занимающие часть окна браузера.

При размещении графики на веб-странице необходимо следить за тем, чтобы размер графических файлов был как можно меньше. Хотя в принципе размеры графических файлов в байтах могут быть неограниченными, следует помнить, что передача большого файла может занять очень много времени, особенно при низкой пропускной способности линий. Если тот, кто смотрит вашу веб-страницу, платит деньги за время нахождения в Интернете, вряд ли он захочет смотреть ваши следующие страницы, учитывая, что первая страница загружается очень долго. В этом уроке будут рассмотрены приемы работы с графикой.

## Размещение графики на веб-странице

Для вставки графики в веб-страницу используется тег `<IMG>`. Графика позволяет значительно улучшить внешний вид и функциональность документов. Тег `<IMG>` (от англ. *image* — изображение) появился в HTML версии 2.0. Он имеет атрибуты `SRC`, `ALT`, `LONGDESC`, `HEIGHT`, `WIDTH`, `USEMAP`, `ISMAP`, `ALIGN`, `BORDER`, `HSPACE`, `VSPACE`.

Пример:

```
<IMG SRC="URL" ALT="текст" HEIGHT="длина" WIDTH="длина"
ALIGN=top|middle|bottom|left|right ISMAP>.
```

- **Атрибут `SRC`.** Атрибут `SRC` (от англ. *source* — источник) определяет месторасположение изображения, включаемого в состав документа. `URL` — обязательный параметр, который указывает браузеру, где находится рисунок. Рисунок должен храниться в графическом формате, например в формате GIF или JPG.

Если графический файл находится в том же каталоге, что и содержащий его HTML-документ, достаточно указать только имя этого файла, например:

```
<IMG SRC=file1.gif>
```

Если графический файл находится на том же сервере, что и содержащий его HTML-документ, но в другом каталоге, следует указать имя каталога и имя этого файла, например:

```
<IMG SRC=picture/file1.gif>.
```

Если графический файл находится не на том сервере, на котором находится содержащий его HTML-документ, необходимо указать полный адрес этого файла, например:

```
<IMG SRC="http://www.uprint.ru/picture/file1.gif">.
```

- **Атрибут `ALT`.** Необязательный атрибут `ALT` задает альтернативный текст, который выводится браузером, пока идет загрузка изображения (или браузером, не поддерживающим отображение графики). Обычно это короткое описание изображения, которое пользователь мог бы или сможет увидеть на экране. Если этот атрибут не задан, на месте изображения большинство браузеров выводят значок, щелкнув на котором, пользователь сможет увидеть изображение. Атрибут `ALT` рекомендуется указывать, если ваши пользователи применяют браузер, не поддерживающий графический режим, например Lunix.
- **Атрибут `LONGDESC`.** Необязательный атрибут `LONGDESC` задает ссылку на удаленный ресурс.
- **Атрибут `HEIGHT`.** Необязательный атрибут `HEIGHT` определяет высоту рисунка в пикселах. Если данный параметр не указан, используется оригинальная высота рисунка. Этот параметр позволяет сжимать или растягивать изображение по вертикали, что дает возможность точнее контролировать внешний вид документа. Однако некоторые браузеры не поддерживают данный параметр. Иногда экранное разрешение на разных компьютерах может быть разным, поэтому при задании абсолютной величины графического объекта следует быть внимательным.
- **Атрибут `WIDTH`.** Необязательный атрибут `WIDTH` используется для указания ширины рисунка в пикселах.

- *Атрибут ALIGN*. Необязательный атрибут **ALIGN** используется для точного позиционирования объектов на экране. Доступные значения:
  - **top** — верхний край объекта выравнивается по верхнему краю строки;
  - **middle** — центр объекта выравнивается по базовой линии строки;
  - **bottom** — нижний край объекта выравнивается по базовой линии строки;
  - **left** — объект выравнивается по левому краю, при этом возможно обтекание объекта текстом;
  - **right** — объект выравнивается по правому краю, при этом возможно обтекание объекта текстом.
- Если данный параметр не указан, большинство браузеров располагает изображение в левой части экрана, а текст — справа от него.
- *Атрибут BORDER*. Атрибут **BORDER** задает толщину обрамления для изображения.
- *Атрибут VSPACE*. Атрибут **VSPACE** позволяет задать размер в пикселах пустого пространства над и под изображением, чтобы текст «не наезжал» на изображение.
- *Атрибут HSPACE*. Атрибут **HSPACE** позволяет задать размер в пикселах пустого пространства слева и справа от изображения, чтобы текст «не наезжал» на изображение.

## Форматы графических файлов

Большинство форматов файлов в компьютерной графике позволяет хранить информацию об изображении как о наборе точек. Точно так же (в виде набора точек) изображения выводятся на экран.

Обычно в Интернете используются два растровых формата — GIF и JPEG. Кроме того, в Интернете довольно много изображений, имеющих форматы BMP и PCX. Специалисты считают, что довольно большие перспективы есть у формата PNG — самого современного формата переносимой сетевой графики.

### Формат GIF

*Формат GIF* (Graphic Interchange Format — формат обмена графическими данными) служит для записи и хранения растровых графических изображений. Этот формат был разработан корпорацией CompuServe с использованием технологии Unisys. Формат GIF применяется для хранения 256-цветных изображений (или изображений с меньшим количеством цветов), сжатых по методу Лемпела—Зива (LZW). Тот же метод сжатия характерен для архиваторов файлов.

Формат GIF дольше других представлен в Интернете и поддерживается разными графическими редакторами. В нем используются индексированные цвета. Версия GIF89a этого формата обеспечивает возможность чересстрочной развертки, что позволяет постепенно повышать качество принятого через канал связи изображения. Это самый распространенный формат изображений в Интернете.

Изображения в формате GIF хранятся в файлах с расширением .gif. К достоинствам GIF-изображений относится то, что вид изображения не зависит от браузера и платформы. Лучше всего отображаются чертежи, рисунки, изображения с небольшим количеством однородных цветов, прозрачные изображения и анимационные последовательности. В GIF-изображениях используется алгоритм сжатия без потери информации.

## Формат JPEG

Применяющийся в *формате JPEG* (Joint Photographic Expert Group — объединенная группа экспертов в области фотографии) алгоритм обработки изображений разработан группой экспертов как средство сжатия изображений с палитрой 24 бита на пиксел, что обеспечивает отображение 16,7 миллиона цветов.

JPEG — один из самых мощных алгоритмов. Практически он является стандартом де-факто для полноцветных изображений. Формат JPEG был создан для того, чтобы избавиться от недостатков формата GIF.

Алгоритм оперирует областями 8×8 пикселей, в которых яркость и цвет плавно меняются. Сжатие в формате JPEG осуществляется за счет того, что в реальных изображениях, в частности в фотографиях, цвет обычно меняется достаточно плавно. Обеспечивается высокий коэффициент сжатия, значение которого достигает 100 и зависит от допустимого уровня потерь изобразительной информации.

Формат широко используется в HTML-документах и для передачи графики по сети. Сохраняет параметры графики в цветовой модели RGB. Изображения в формате JPEG хранятся в файлах с расширением .jpg.

В формате JPEG используются алгоритмы сжатия с потерей информации. При сжатии из изображения исключаются данные, которые считаются несущественными. При этом есть риск получить нечеткое, размытое изображение с искажением деталей.

## Формат PNG

Последнее время получили распространение файлы в *формате PNG* (Portable Network Graphics — переносимая сетевая графика). Само название формата говорит о его назначении — использовании при передаче изображений в сетях. Формат поддерживает полноцветные RGB-изображения и индексированные изображения. Допускает наличие дополнительного канала для хранения маски обрезки. Имеет эффективный алгоритм сжатия без потери информации.

## Формат BMP

Первым графическим форматом для IBM-совместимых компьютеров, нашедшим широкое применение, был *формат BMP* (Bit Map — битовая карта). Можно сказать, что с него все началось. В формате BMP первоначально использовалось простейшее кодирование — по пикселям (самое неэкономное). Пиксели обходились последовательно по строкам, начиная с нижнего левого угла графического изображения. Файлы именно этого формата поддерживались первыми версиями

оболочки Windows. В них использовались только индексированные цвета в количестве 256, то есть один пиксел представлялся одним байтом. В дальнейшем формат стал применяться и для кодирования полноцветных изображений. Это стандартный формат растровой графики. Поскольку BMP-файлы представляют собой «родной» формат графики для Microsoft Windows, подобных изображений в Интернете достаточно много.

## Формат PCX

*Формат PCX* использует только индексированные палитры. Он был разработан фирмой Z-soft специально для пакета PaintBrush, а в дальнейшем получил более широкое распространение. В нем предусмотрено сжатие, обеспечивающее большую компактность по сравнению с форматом BMP. Тем не менее практически он пригоден только для простых изображений или изображений с индексированными цветами.

## Пикселы и разрешение

В Интернете для передачи изображений в основном используется растровая графика.

*Растровая графика* — это, во-первых, графика, представляемая в компьютере в виде множества точек (пикселов), во-вторых, файл определенного формата.

Для растровой графики важной характеристикой является разрешение изображения. Одна и та же картинка может быть представлена с лучшим или худшим качеством в соответствии с количеством точек на единицу длины.

*Разрешение* — количество точек на единицу измерения:

- dpi (dots per inch) — количество точек на дюйм;
- ppi (points/pixels per inch) — количество пикселов на дюйм;
- spi (samples per inch) — оптическое, первоначальное разрешение сканера, означающее количество проб на дюйм, то есть число «просмотров» сканером изображения во время сканирования.

Dpi — наиболее часто используемый термин, который относится только к маленьким точкам, создаваемым выходным устройством. Точки выходного устройства ничего общего не имеют с разрешением изображения, они связаны с минимальным воспроизводимым принтером размером точки.

Разрешение иллюстрации обычно измеряется в dpi. Чем больше разрешение, тем более качественным является изображение, но и тем больший размер имеет файл, в котором изображение сохраняется.

Различают разрешение изображения, разрешение экрана и разрешение принтера.

Мониторы, выпускаемые различными фирмами, как правило, имеют стандартные разрешения: 640×470, 700×600, 1024×768, 1270×1024, 1600×1270, 1920×1600 точек. Расстояние между люминофорами хорошего монитора составляет 0,2–0,25 мм. Конечно, для качественного дизайна требуется монитор с большим разрешени-

ем. Иногда дизайнер может редактировать изображение попиксельно. Разрешающая способность экрана зависит от монитора и видеоадаптера компьютера.

Статические растровые изображения представляют собой двухмерный массив чисел. Элементы этого массива называют *пикселями* (от англ. pixel — picture element, или элемент изображения). Пиксел — минимальный адресуемый элемент двухмерного растрового изображения; одна точка изображения.

## Примеры размещения графики на веб-странице

Прежде чем размещать изображение на веб-странице, проведем эксперимент. Возьмем любительскую фотографию высотой 11 см и шириной 13 см. Как она будет выглядеть на экране? Если мы отсканируем фотографию в трех вариантах, в первом выберем цветной режим с разрешением 300 dpi, во втором — черно-белый режим с разрешением 300 dpi, в третьем — цветной режим с разрешением 72 dpi, то в первых двух вариантах отсканированная фотография будет иметь размер 1304×1479 пикселей, а в третьем варианте — 313×355 пикселей.

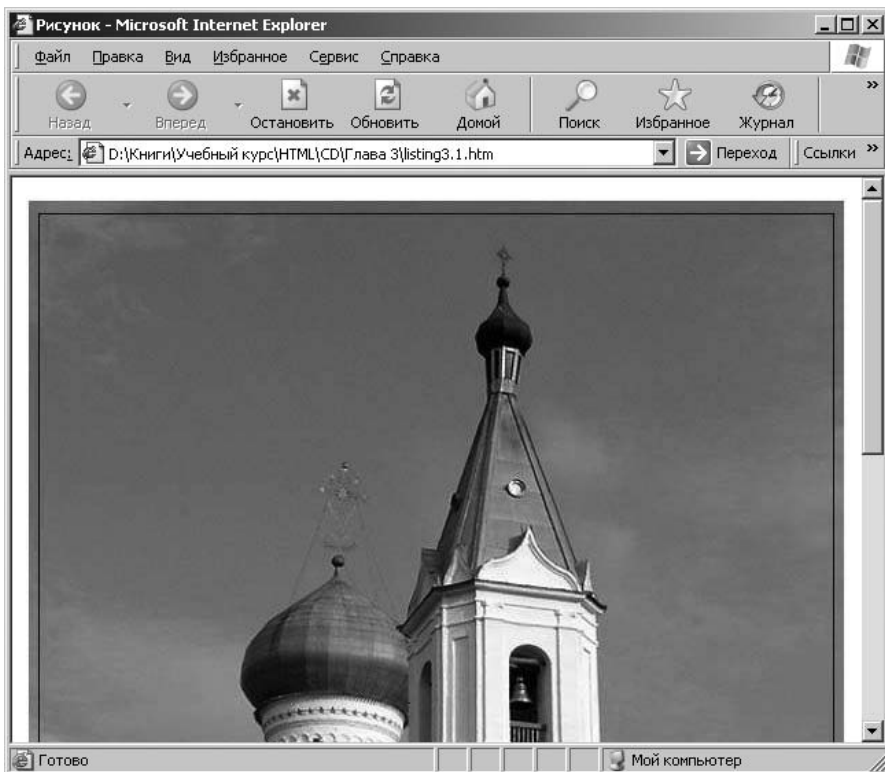


Рис. 3.1. Купола на веб-странице

Поскольку размеры фотографии в пикселах в первых двух вариантах оказались больше размера экрана в пикселах (размер, например, моего экрана составляет 768×1024 пикселей), при размещении фотографии на веб-странице для ее просмотра потребуются полосы прокрутки. В третьем варианте отсканированная фотография занимает только часть экрана.

Таким образом, для изображений, распространяемых через Интернет, обычно устанавливается разрешение 72 dpi, а разрешение в 300 dpi характерно больше для печатных изображений.

На рис. 3.1 показан пример размещения на веб-странице произвольного изображения (отсканированной фотографии), соответствующий код приведен в листинге 3.1.

### Листинг 3.1. Пример создания веб-страницы с изображением церкви

```
<HTML>
<HEAD>
<TITLE> Рисунок</TITLE>
</HEAD>
<BODY>
<IMG SRC="Церковь.jpg">
</BODY>
</HTML>
```

В этом примере при размещении изображения на странице мы не касались вопросов его размеров и положения на странице. На самом деле эти вопросы необходимо учитывать.



#### СОВЕТ

Размеры изображения можно уточнить и при необходимости изменить в программе Adobe Photoshop.

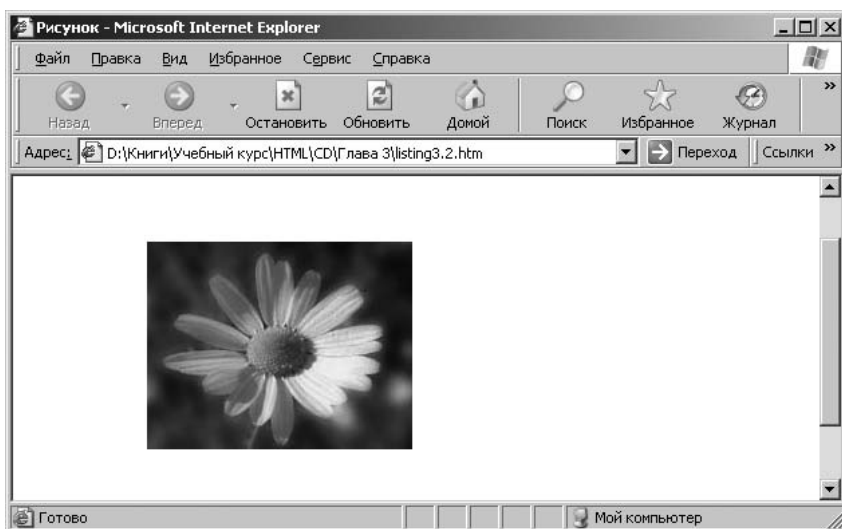


Рис. 3.2. Изображение ромашки на веб-странице

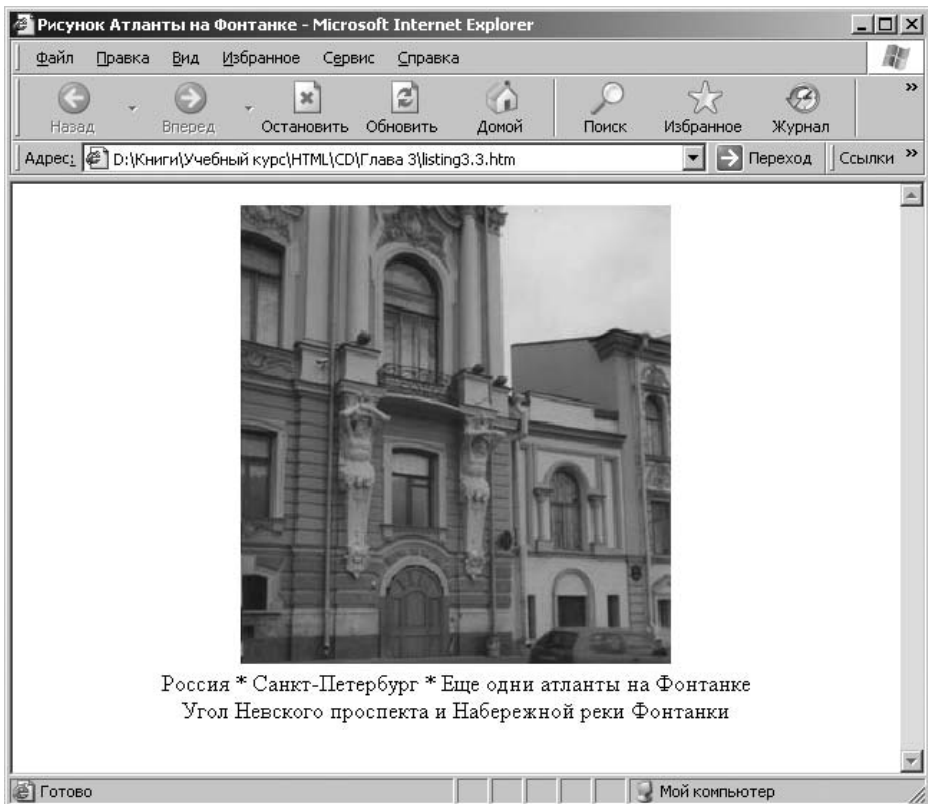
На рис. 3.2 показан пример размещения изображения шириной 198 и высотой 155 пикселей (листинг 3.2). Хотя листинги 3.1 и 3.2 практически совпадают, изображения имеют разные размеры в пикселях и поэтому занимают разную площадь.

**Листинг 3.2.** Пример создания веб-страницы с изображением ромашки

```
<HTML>
<HEAD>
<TITLE> Рисунок</TITLE>
</HEAD>
<BODY TOPMARGIN=100 LEFTMARGIN=100>
<IMG SRC="Ромашка.jpg">
</BODY>
</HTML>
```

По умолчанию рисунок разместился в левом верхнем углу.

На рис. 3.3 показан еще один пример размещения на веб-странице изображения, но на этот раз в коде заданы значения атрибутов ширины и высоты (ширина 281, высота 300 пикселей), а также указана подпись (листинг 3.3).



**Рис. 3.3.** Веб-страница с изображением и подписью

**Листинг 3.3.** Размещение на веб-странице изображения и подписи

```

<HTML>
<HEAD>
<TITLE>Рисунок Атланты на Фонтанке</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="#000000">
<CENTER>
<IMG WIDTH ="300" HEIGHT="281"
SRC="Атланты на Фонтанке.jpg">
<BR>
Россия * Санкт-Петербург*Еще одни атланты на Фонтанке <BR>
Угол Невского проспекта и Набережной реки Фонтанки
</CENTER>
</BODY></HTML>

```

На рис. 3.4 показан пример размещения четырех изображений шириной 92 и высотой 114 пикселей каждое в таблице (листинг 3.4). Подробно принципы работы с таблицами будут рассмотрены далее.



**Рис. 3.4.** Четыре рисунка в таблице

**Листинг 3.4.** Пример создания четырех рисунков в таблице

```

<HTML><HEAD><TITLE>Четыре рисунка в таблице</TITLE></HEAD>
<BODY bgColor=#A1633D>
<TABLE BORDER=0 CELLpADDING=1 CELLSPACING=0>
<TR>
<TD>
<IMG SRC="Лена.jpg" WIDTH="92" HEIGHT="114"> </TD>
<TD>
<IMG SRC="Подснежник.jpg" WIDTH="92" HEIGHT="114"> </TD></TR>
<TR>
<TD><IMG SRC="Цветы.jpg" WIDTH="92" HEIGHT="114">
</TD>
<TD><IMG SRC="Лена.jpg" WIDTH="92" HEIGHT="114">
</TD>
</TR>
</BODY></HTML>

```

## Выравнивание изображений

С помощью атрибута ALIGN можно задать способ выравнивания изображения на веб-странице. На рис. 3.5 первый рисунок выровнен вправо, второй — влево (листинг 3.5).

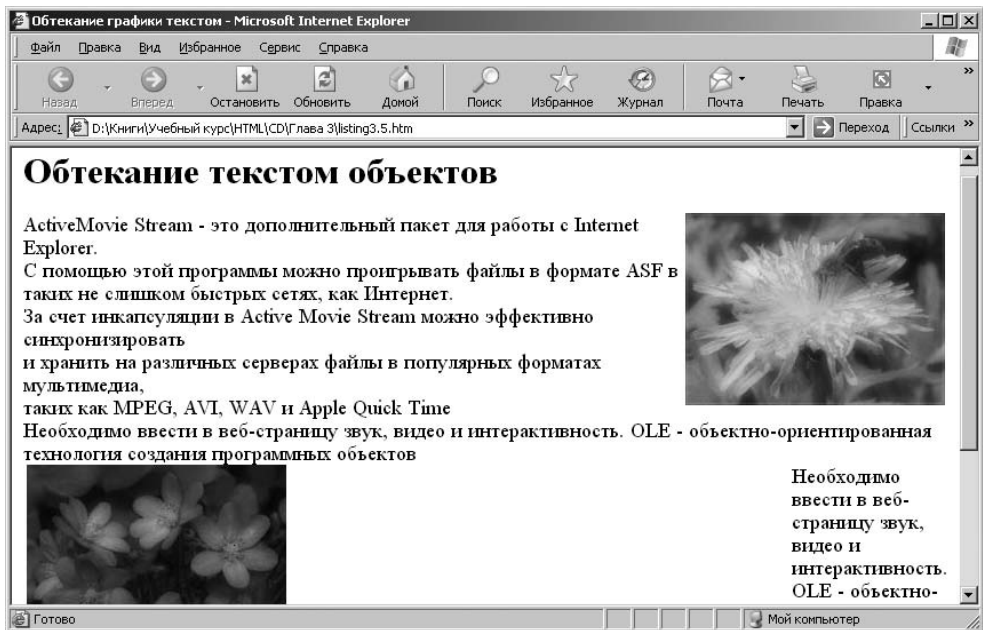


Рис. 3.5. Пример выравнивания изображений на веб-странице

**Листинг 3.5. Пример выравнивания изображений на веб-странице**

```

<HTML><HEAD>
<TITLE> Обтекание графики текстом</TITLE>
</HEAD>
<BODY>
<H1>Обтекание текстом объектов</H1>
<IMG SRC="Одуванчик.jpg" ALIGN=RIGHT>
<FONT size=4>ActiveMovie Stream - это дополнительный пакет для работы с Internet
  Explorer.<BR>С помощью этой программы можно проигрывать файлы в формате ASF в таких
  не слишком быстрых сетях, как Интернет.<BR>
  За счет инкапсуляции в Active Movie Stream можно эффективно синхронизировать <BR>
  и хранить на различных серверах файлы в популярных форматах мультимедиа,<BR>
  таких как MPEG, AVI, WAV и Apple Quick Time<BR>
  Необходимо ввести в веб-страницу звук, видео и интерактивность. OLE -
  объектно-ориентированная технология создания программных объектов
<IMG SRC="Печеночник.jpg" ALIGN=LEFT><BR>
<TABLE BORDER=2 ALIGN=RIGHT>
<TR>
<TD>Возможно </TD>
</TR>
<TR>
<TD>обтекание текста </TD>
</TR>
<TR>
<TD>вокруг таблицы </TD>
</TR>
  Необходимо ввести в веб-страницу звук, видео и интерактивность. OLE -
  объектно-ориентированная технология создания программных объектов </FONT>
</BODY></HTML>

```

## Карты изображений

*Карты изображений* (image maps) позволяют пользователю связывать ссылки на другие документы с отдельными частями изображений. Щелкая мышью на частях изображения, так называемых *активных областях* (hot spots), пользователь может выполнять те или иные действия, переходить к другим документам и т. п.

Если пользователь позиционирует указатель мыши на изображении, которое с помощью атрибута ISMAP в теге <IMG> определено как карта изображения, и щелкает левой кнопкой мыши, то координаты точки щелчка передаются на веб-сервер. При получении этих координат сервер ищет на карте активную область, содержащую эти координаты. Если такая область существует, то заданный URL-адрес активизируется, и браузер пользователя переходит по этому адресу на новую страницу.

Описанный вариант функционирования характерен для случая, когда карта изображения размещается на сервере, однако она также может размещаться на клиентском компьютере. В этом случае браузер самостоятельно выясняет URL-адрес нужной страницы и передает серверу не координаты точки щелчка, а этот адрес.

Второй случай предпочтительнее, так как на сервере не приходится запускать программу (сценарий), выполняющую просмотр изображения с целью определения активной области. По соображениям безопасности многие системные адми-

нистраторы не разрешают посторонним работать на сервере и записывать сценарии в каталоги.



## ВНИМАНИЕ

К сожалению, с картами изображений, функционирующими на клиентской машине, могут работать не все браузеры.

## Тег <MAP>

Тег <MAP> применяется для представления графического изображения в виде карты с активными областями. Активные области можно идентифицировать по всплывающей подсказке, которая появляется при наведении на такую область указателя мыши. Данный тег появился в HTML версии 3.2.

Контейнер, состоящий из тегов <MAP> и </MAP>, является блоком описателей активных областей.

Тег <MAP> может быть связан с тегом <IMG> или тегом <INPUT> с помощью атрибута USEMAP.

Чтобы включить поддержку карты изображения, необходимо ввести атрибут USEMAP в тег <IMG>:

```
<IMG SRC="URL" USEMAP="URL#map_name">
```

Атрибут USEMAP указывает, в каком месте находится карта изображения. Карта изображения определяет, какому участку изображения какой URL-адрес соответствует. Карта изображения может находиться в том же документе, что и изображение, или в другом документе. Размещение карты в другом документе позволяет объединять вместе изображения из разных документов, но добавляет еще одну итерацию в механизм функционирования карты, поскольку за первую итерацию выясняется местонахождение карты, а за вторую — выполнение действия, предписанного URL-адресом данного участка изображения. Параметр map\_name указывает имя карты для изображения, а предшествующий ему URL-адрес — местонахождение карты. Если параметр URL отсутствует, карта с указанным именем ищется в текущем документе.

Пример синтаксиса определения карты изображения:

```
<MAP NAME="map_name"> <AREA [SHAPE=" shape "] COORDS="x,y,..." [HREF=" reference "] [NOHREF]></MAP>.
```

Атрибут NAME задает имя карты, например: NAME="КАРТА" — имя карты.

## Тег <AREA>

Тег <AREA> определяет активную область в составе карты изображений на стороне клиента (это описатель активной области). Тег <AREA> не имеет закрывающего тега. Синтаксис:

```
<MAP name="имя"><AREA атрибуты></MAP>
```

Для каждой активной области карты имеется свой тег <AREA>.

○ *Атрибут SHAPE.* Атрибут SHAPE задает форму активной области на карте и ее координаты, он может принимать значения:

□ rect — прямоугольная область;

- `circle` — область в форме круга;
- `poly` — область в форме многоугольника;
- `default` — вся область.

Атрибуты, определяющие форму области на карте, являются обязательными.

- *Атрибут COORDS.* Атрибут `COORDS` определяет позицию области на экране. Синтаксис:

```
COORDS="координаты"
```

Набор координат соответствует форме области. Координаты задаются в единицах длины и разделяются запятыми:

- для круга необходимо задать координаты центра и радиус в пикселах (координаты центра отсчитываются от левого края и верхнего края рисунка);
  - для прямоугольной области задаются координаты всех четырех углов;
  - для многоугольной области задаются координаты каждого из углов многоугольника, который точно или приблизительно соответствует форме области.
- *Атрибут HREF.* Атрибут `HREF` задает URL-адрес для ссылки. Синтаксис:

```
<AREA HREF="URL">
```

- *Атрибут ALT.* Атрибут `ALT` задает альтернативный текст для браузеров, которые не поддерживают тег `<AREA>`. Синтаксис:

```
<AREA ALT = "альтернативный текст">
```

- *Атрибут TITLE.* Атрибут `TITLE` задает текст подсказки, всплывающей при наведении указателя мыши на активную область. Синтаксис:

```
<AREA TITLE = "текст подсказки">
```

- *Атрибут TARGET.* Атрибут `TARGET` идентифицирует фрейм, в котором должен быть открыт целевой ресурс. Синтаксис:

```
TARGET="имя|_blank|_self|_parent|_top"
```

Параметры:

- `имя` — документ открывается во фрейме с именем, задаваемым с помощью атрибута `NAME` дескриптора `<FRAME>` (имя фрейма должно начинаться с буквы);
  - `_blank` — документ открывается в новом неименованном окне;
  - `_self` — документ открывается в том же окне, в котором содержится текущий документ;
  - `_parent` — документ открывается во фрейме, родительском по отношению к фрейму, содержащему текущий документ (если родительского фрейма не существует, документ открывается в текущем фрейме);
  - `_top` — документ открывается в текущем окне и занимает все окно.
- *Атрибут NOHREF.* Атрибут `NOHREF` указывает на то, что с областью не связана ссылка.
  - *Атрибут TABINDEX.* Атрибут `TABINDEX` определяет порядок, в котором активные области получают фокус после нажатия клавиши `Tab`. Синтаксис:

```
TABINDEX="число"
```

- *Атрибут ACCESSKEY*. Атрибут ACCESSKEY определяет комбинацию клавиш, используемую для доступа к активной области (перехода по URL). Синтаксис:

```
ACCESSKEY="символ"
```

- *Атрибут ONFOCUS*. Атрибут ONFOCUS определяет обработчик события, возникающего при получении активной областью фокуса ввода (фокус ввода может быть получен как с помощью мыши, так и в результате перебора после нажатия клавиши Tab). Синтаксис:

```
ONFOCUS="сценарий"
```

- *Атрибут ONBLUR*. Атрибут ONBLUR определяет обработчик события, возникающего при потере активной областью фокуса ввода (фокус ввода может быть потерян как в результате выбора другого элемента с помощью мыши, так и в результате перебора после нажатия клавиши Tab). Синтаксис:

```
ONBLUR="сценарий"
```

- *Атрибут USEMAP*. Необязательный атрибут USEMAP связывает с данным элементом карту изображения. Синтаксис:

```
USEMAP="URL"
```

Этот атрибут разрешает использовать изображения, отдельные части которого связаны с картами и объединяются тегом <MAP>. В этом случае щелчок на карте обрабатывает браузер:

```
<IMG SRC = "URL изображения" USEMAP = "URL" карты (# имя карты) >
```

- *Атрибут ISMAP*. Необязательный атрибут ISMAP используется для определения карты изображения на стороне сервера. Синтаксис:

```
ISMAP="длина"
```

Тег <IMG> должен быть включен в тег <A>. Атрибут ISMAP сообщает браузеру, что данное изображение дает пользователю возможность выполнять какие-либо действия, щелкая мышью на активных областях. Этот атрибут разрешает использовать изображения, отдельные части которого связаны со ссылками и позволяют выполнять переходы. Такие изображения применяются совместно с тегом <A>. В этом случае реакцию на щелчок обрабатывает сервер:

```
<A HREF = "URL карты"> <IMG SRC = "URL изображения" ISMAP>
```

## Примеры карт изображений

В качестве примера карты изображения рассмотрим семейную фотографию. При наведении указателя мыши на каждого члена семьи появляется соответствующая всплывающая подсказка и возможен переход на другую веб-страницу с более подробными данными о нем.

Код карты представлен в листинге 3.6, а на рис. 3.6 показаны варианты карты изображения с активными областями в виде круга, прямоугольника и многоугольника. Под изображением находится поле, в котором появляется название области при наведении на нее указателя мыши. Изображение, которое выбрано для примера, имеет размер 400×300 пикселей.



Рис. 3.6. Пример карты изображения

**Листинг 3.6.** Пример создания карты изображения с активными областями в виде круга, прямоугольника, многоугольника

```
<HTML><HEAD><TITLE>Фотография-карта</TITLE></HEAD>
<BODY BGCOLOR=#FFFFFF>
<IMG HEIGHT=300 SRC="Листинг 3.6.files/Семья.gif" width=400 USEMAP=#Map BORDER=0>
<MAP NAME=Map>
<AREA onmouseover="document.where.who.value='Папа'"
SHAPE=CIRCLE COORDS=70,70,70 HREF="Father.htm">
<AREA onmouseover="document.where.who.value='Сынок'"
SHAPE=POLY COORDS=130,50,250,50,270,156,260,196,220,290,140,141
HREF="Son.htm">
<AREA onmouseover="document.where.who.value='Дочка'"
SHAPE=RECT COORDS=59,125,170,220 HREF="Daughter.htm">
<AREA onmouseover="document.where.who.value='Мама'"
```

```

SHAPE=CIRCLE COORDS=310,80,70 HREF="Mather.htm">
</MAP>
<FORM NAME=where><INPUT size=30 NAME=who> </FORM>
</MAP>
</BODY></HTML>

```

В данном примере изображение, размещенное на странице с помощью тега <IMG>, становится картой с помощью атрибута USEMAP.

В листинге 3.7 представлен код еще одного примера карты изображения, позволяющей просмотреть более подробно пригороды Санкт-Петербурга, выбирая часть большой фотографии. Фотография разделена на четыре прямоугольника (рис. 3.7).

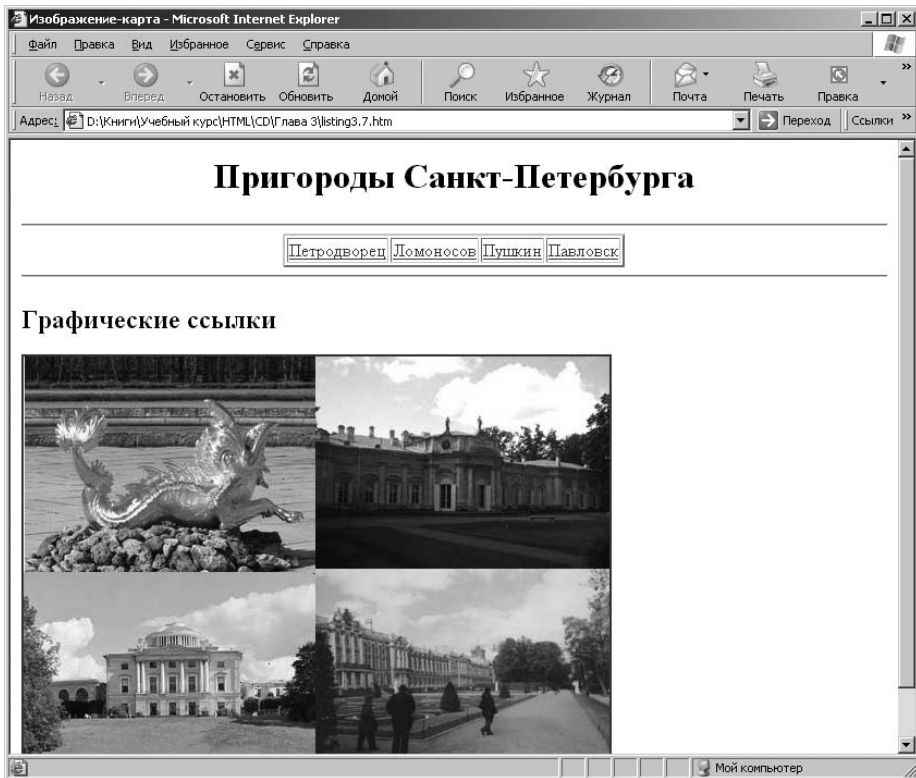


Рис. 3.7. Еще один пример карты изображения

**Листинг 3.7.** Пример создания карты изображения с активными областями в виде прямоугольников

```

<HTML><HEAD><TITLE>Меню - карта</TITLE></HEAD>
<BODY BGCOLOR = #FFBCBC>
<CENTER>
<H1>Пригороды Санкт-Петербурга</H1>
<HR>
<TABLE BORDER=2>

```

```

<TR><TD><A HREF = graph1.htm> Петродворец</A>
<TD><A HREF = #img>Ломоносов</A>
<TD><A HREF = move.htm> Пушкин</A>
<TD><A HREF = #map>Павловск</A>
</TABLE><HR></CENTER>
<A NAME = MAP>
<H2>Графические ссылки</H2>
<MAP NAME = "mymap">
<AREA SHAPE = "RECT" COORDS = "0,0,270,190" HREF = Петродворец.htm>
<AREA SHAPE = "RECT" COORDS = "270,0,540,190" HREF = Ломоносов.htm>
<AREA SHAPE = "RECT" COORDS = "0,190,270,380" HREF = Павловск.htm>
<AREA SHAPE = "RECT" COORDS = "270,190,540,380" HREF = Пушкин.htm>
</MAP>
<IMG SRC = "Пригороды.jpg" USEMAP = "#mymap">
<HR>
<A NAME = IMG>
</BODY>
</HTML>

```

Чтобы лучше понять, как задаются координаты областей, приведен рис. 3.8 с координатами в пикселах для каждой прямоугольной области.



Рис. 3.8. Разметка прямоугольных областей карты

Это изображение, состоящее из четырех видов пригородов Санкт-Петербурга, имеет размеры 380×540 пикселей.

## Масштабирование изображений

На рис. 3.9 показана веб-страница (листинг 3.8), содержащая три изображения разного размера, источником которых является один и тот же файл. Файл имеет размеры 154 пиксела в ширину и 248 пикселей в высоту. Первое изображение

увеличено в два раза, второе имеет реальный размер, размеры третьего изображения уменьшены в два раза.

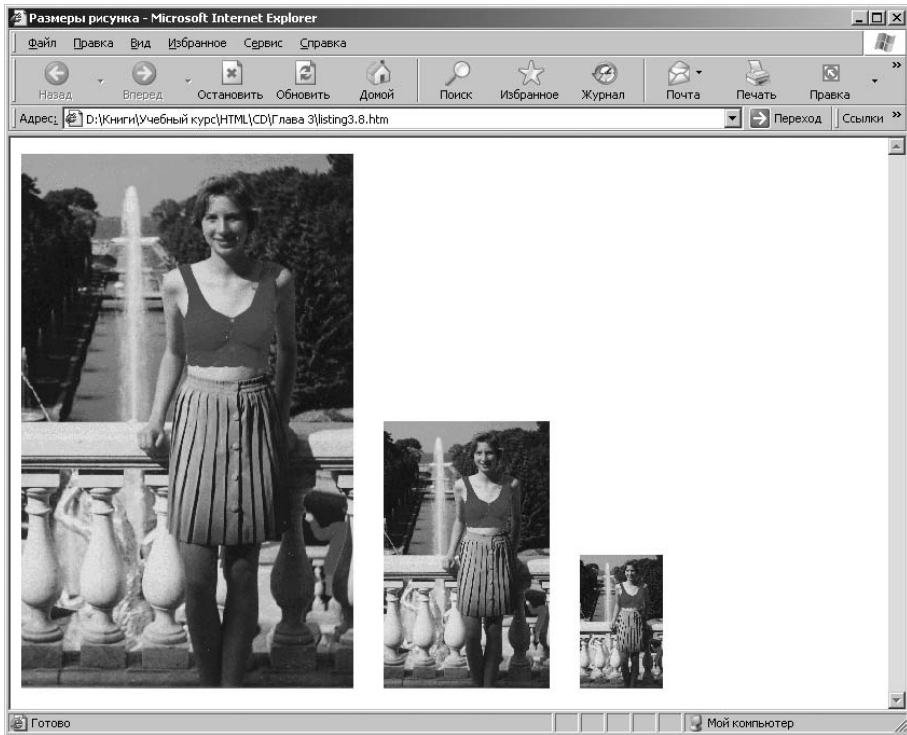


Рис. 3.9. Три изображения разного размера из одного файла

**Листинг 3.8.** Пример создания трех изображений разного размера из одного файла

```
<HTML><HEAD><TITLE>Размеры рисунка</TITLE></HEAD>
<BODY>
<IMG Width=308 Height= 496 src="girl.gif"> &nbsp; &nbsp; &nbsp; &nbsp;
<IMG Width=154 Height= 248 src="girl.gif"> &nbsp; &nbsp; &nbsp; &nbsp;
<IMG Width=77 Height= 124 src="girl.gif">
</BODY></HTML>
```

Изображения отстоят друг от друга на расстояние трех неразрывных пробелов (которые задаются символами &nbsp;).

Если непродуманно задать ширину и высоту изображения, оно исказится. Такой пример приведен на рис. 3.10 (листинг 3.9).

**Листинг 3.9.** Пример неудачного получения трех изображений разного размера из одного файла

```
<HTML><HEAD><TITLE>Искаженные рисунки</TITLE></HEAD>
<BODY><IMG height=424 width=376 src="girlie.gif"> &nbsp; &nbsp; &nbsp; &nbsp;
<IMG height=400 width=154 src="girlie.gif"> &nbsp; &nbsp; &nbsp; &nbsp;
<IMG height=124 width=376 src="girlie.gif"> </BODY></HTML>
```

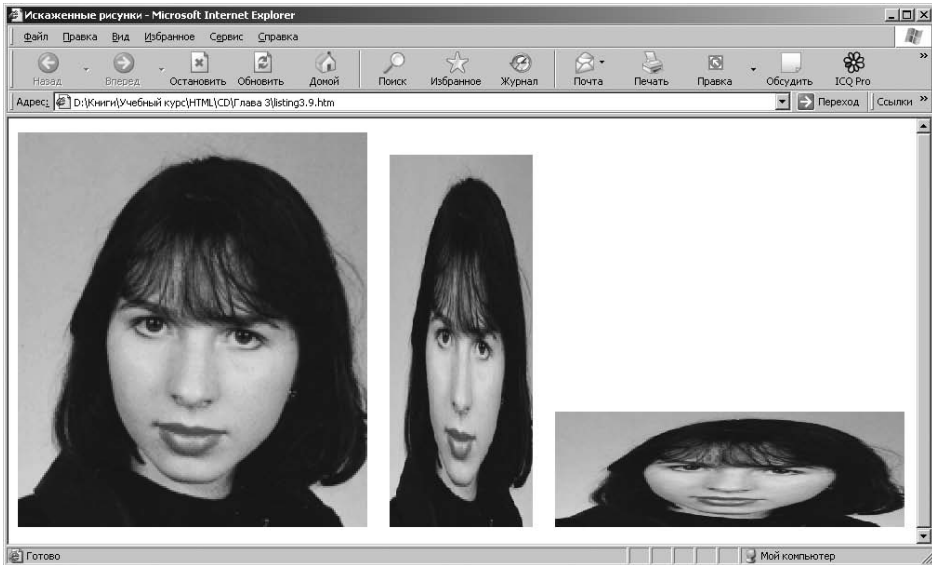


Рис. 3.10. Три изображения разного размера, полученные из одного файла

Стандартные современные мониторы имеют разрешение 1024 пиксела в ширину и 768 пикселей в высоту. При разработке изображений следует учитывать, что окно просмотра будет занимать лишь часть этой области. На рис. 3.11 показаны изображения в виде полос с указанием их ширины и высоты в пикселах (листинг 3.10). Первая полоса имеет размеры 90 пикселей в высоту и 980 в ширину, вторая — 80 пикселей в высоту и 780 в ширину, третья — 75 пикселей в высоту и 600 в ширину, четвертая — 70 пикселей в высоту и 500 в ширину, пятая — 65 пикселей в высоту и 400 в ширину, шестая — 60 пикселей в высоту и 200 в ширину, седьмая — 50 пикселей в высоту и 100 в ширину.

### Листинг 3.10. Примеры создания изображений с указанием их размеров в пикселах

```
<HTML><HEAD><TITLE>Размеры рисунков в пикселах</TITLE>
</HEAD>
<BODY>
<IMG height=90 width=980 src="poly1.gif" > <BR>
<IMG height=80 width=780 src="poly2.gif"> <BR>
<IMG height=75 width=600 src="poly3.gif"> <BR>
<IMG height=70 width=500 src="poly4.gif"> <BR>
<IMG height=65 width=400 src="poly5.gif"> <BR>
<IMG height=60 width=200 src="poly6.gif"> <BR>
<IMG height=50 width=100 src="poly7.gif">
</BODY></HTML>
```

Если размер изображения по ширине превышает 980 пикселей, оно не может полностью поместиться на экране, и появляется горизонтальная полоса прокрутки. Если размер изображения по высоте превышает 570 пикселей, оно не может полностью поместиться на экране, и появляется вертикальная полоса прокрутки. Для мониторов, размеры которых в пикселах отличаются от величины

1024×768, размеры полностью размещаемого на экране изображения будут другие, но суть останется та же.

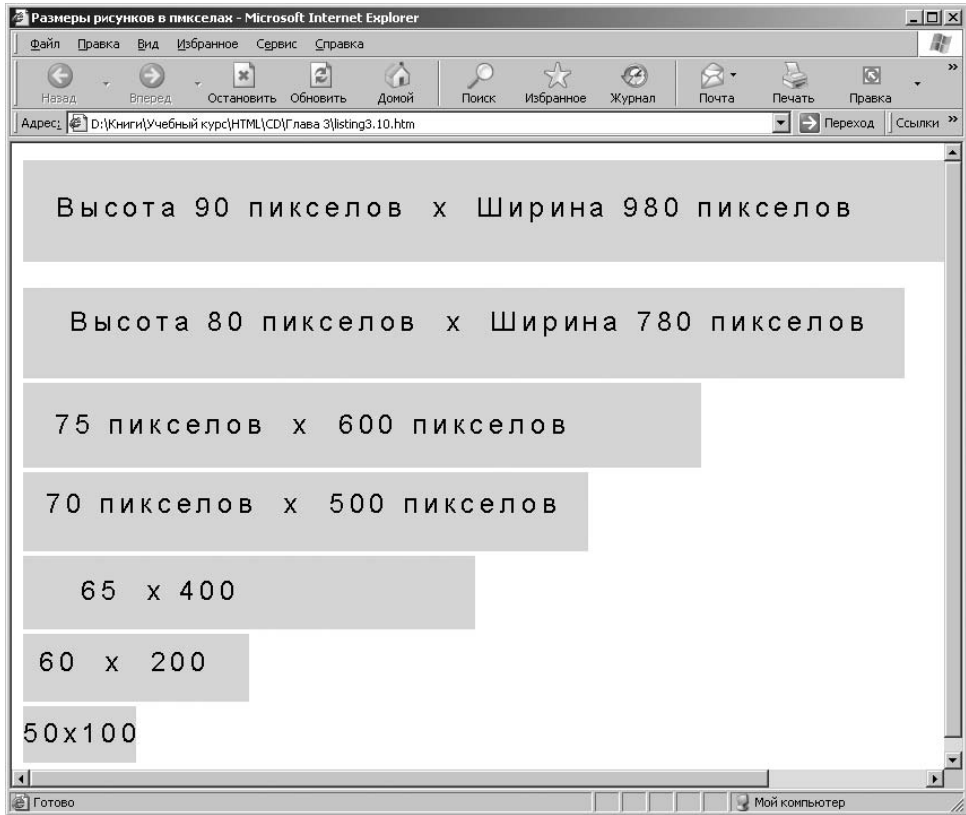


Рис. 3.11. Примеры изображений с указанием их размеров в пикселях

## Подведем итоги

В этом уроке рассмотрены теги, позволяющие размещать графику на веб-страницах. Главным из них является тег `<IMG>` с его атрибутами `SRC`, `ALT`, `LONGDESC`, `HEIGHT`, `WIDTH`, `ALIGN`, `BORDER`, `VSPACE`, `HSPACE`. Местоположением изображения на странице управляет атрибут выравнивания `ALIGN`, который может принимать значения `top`, `middle`, `bottom`, `left` и `right` (сверху, посередине, снизу, слева, справа). Рассказано также о теге `<MAP>`, который применяется для представления графического изображения в виде карты с активными областями. Представлены примеры карт.

Рассмотрены форматы графических файлов JPEG и GIF, которые обычно используются в Интернете. Рассказано также о форматах BMP, PCX и PNG.

Для лучшего понимания природы изображений раскрыты такие понятия, как «пиксели» и «разрешение».