

## Лекция. Изображения в HTML

В HTML предусмотрен целый ряд возможностей для работы с мультимедиа. Это встраивание графики и использование звуков, применение анимационных роликов и видеофильмов. Хорошо оформленная web-страница позволит привлечь и удержать внимание посетителей, в то же время чрезмерное увлечение изображениями, звуковыми эффектами и т.п. может затруднить просмотр страницы (при медленных каналах связи) или отпугнуть чересчур навязчивыми мелодиями.

В общем случае при использовании мультимедиа на web-страницах желательно придерживаться следующих правил: оформление сайта должно соответствовать его содержанию, применение графики и мультимедиа должно упрощать навигацию по сайту, должна быть предусмотрена возможность работы с web-страницей для медленных каналов связи (замещающие надписи, дублирующие текстовые меню или специальные текстовые варианты страницы).

### Графические форматы

Изначально HTML поддерживал работу с двумя графическими форматами: GIF и JPEG, в последнее время все большее распространения находит формат PNG. Каждый из форматов имеет свои достоинства и недостатки, которые определяют область его применения.

#### GIF – Graphics Interchange Format

Формат GIF был разработан коммерческой Online-службой CompuServe для передачи графики по сети между различными платформами. GIF поддерживает до 256 цветов, что является существенным ограничением формата. GIF практически непригоден для передачи фотографических изображений.

В то же время формат GIF обладает рядом свойств, определивших его широкое применение при разработке Web-страниц:

- Прозрачность. Изображения в формате GIF могут иметь прозрачные области, что позволяет отойти от стандартных прямоугольных форм и создавать элементы страницы произвольной формы.
- Чередование строк. Графический файл может быть записан таким образом, что при отображении сначала будет загружаться каждая четвертая строка, растянутая на три соседних. Естественно, качество отображения будет достаточно низкое. Затем, по мере загрузки последующих строк, изображение примет окончательный вид. Такая схема позволяет пользователю еще до окончательной загрузки графического файла понять, что именно представлено на изображении.
- Анимация. Форматом GIF предусмотрена возможность создания анимированных изображений. Фактически это несколько изображений, записанных в один файл и чередующихся с заданным интервалом времени.

Область применения GIF-файлов ограничивается и применяемым алгоритмом сжатия изображений, чья эффективность зависит от разнообразия цветов в изображении, наличия одноцветных областей. Наибольшую экономию объема файла дают изображения с небольшим количеством цветов и большими одноцветными областями. Поэтому формат GIF широко используется для отображения графических меню и кнопок, а также для создания технических иллюстраций (графики, схемы, диаграммы и т.п.). В то же время использование формата GIF для отображения полноцветных фотографий практически не находит применения по причине малого количества передаваемых цветов и низкой эффективности алгоритма сжатия для таких изображений.

## JPEG – Joint Photographic Experts Group

Формат JPEG был разработан для передачи фотографий между различными платформами. Благодаря поддержке 24-битовой цветовой палитры (миллионы цветов) формат JPEG получил широкое распространение для отображения иллюстраций высокого качества. При этом применяемый в JPEG алгоритм сжатия позволяет варьировать размер загружаемого файла в зависимости от требуемого качества отображения иллюстрации.

Кроме того, формат JPEG поддерживает опцию прогрессивный JPEG, которая во многом похожа со свойством чередования строк формата GIF, но реализована на совершенно других принципах. В файл JPEG записывается несколько иллюстраций полного размера, но различного уровня качества. Сначала отображается рисунок низкого качества (и, соответственно, малого объема), затем догружаются все более качественные изображения. При этом общий объем загружаемого файла возрастает несущественно по сравнению с оригинальным JPEG.

## PNG – Portable Network Graphics

В последнее время все большее распространение получает формат PNG, который был разработан для сетевых иллюстраций с целью заменить форматы JPEG и GIF и объединивший в себе все их преимущества.

Формат PNG поддерживает 24-, 32- и 48-битовую цветовую палитру, 8- и 16- битовую палитру серого, 8-битовую пользовательскую палитру. Реализованный в PNG алгоритм сжатия позволяет сжимать изображения лучше чем GIF (от 5 до 25%).

## Изображения в HTML. Размещение изображений

Для размещения изображений в HTML используется тег `<img>` с использованием обязательного атрибута `src` (от source – источник), который указывает URL-адрес загружаемого файла с изображением.

Синтаксис определения изображения:

```

```

Например:

- `<IMG SRC=picture.gif>` - в HTML-документ будет помещено изображение, находящееся в файле `picture.gif`;
- `<IMG SRC=Images/Плитка.jpeg>` - в HTML-документ будет помещено изображение, находящееся в файле `Плитка.jpeg`, который расположен в папке `Images`, находящейся в этой же самой папке, что и HTML-документ.

URL указывает на то место, где хранится изображение. В следующем примере на Web-странице выводится изображение логотипа, расположенное в каталоге "img" на сервере "www.mggtk.ru" (URL: <http://www.mggtk.ru/img/logo.gif>).

```
<html>
<body>

</body>
</html>
```

Браузер размещает изображение там, где в документе появляется тег `<img>`. Если разместить тег `<img>` между двумя абзацами, то браузер покажет первый абзац, затем изображение, а затем второй абзац.

```
<html>
<body>
<p>Первый абзац</p>

<p>Второй абзац</p>
</body>
</html>
```

Если же разместить тег `<img>` внутри абзаца, то изображение будет отображено в одной строке с текстом.

```
<html>
<body>
<p>В этом примере изображение

размещено прямо внутри абзаца</p>
</body>
</html>
```

Формат изображения, его расположение и ряд других параметров задается при помощи атрибутов тега `<img>`.

### Выравнивание изображений

Для выравнивания изображений используется атрибут `align`.

- `align=top` – изображение выравнивается по верхнему краю текущей текстовой строки, не меняя позиции по горизонтали. При этом речь идет как о тексте, так и о графике;
- `align=middle` – изображение центрируется по вертикали на базовой линии текущей текстовой строки, не меняя позицию по горизонтали;
- `align=bottom` – нижний край изображения выравнивается по горизонтали на базовой линии текущей текстовой строки;
- `align=left` – изображение смещается к левому краю рабочей зоны, последующий текст сразу же начинает "обтекать" графику;
- `align=right` – то же что и для `left`, только изображение смещается к правой части рабочей зоны.

Следующий пример показывает различные варианты выравнивания изображений в тексте.

```
<html>
<body>
<p>Изображение выравнено по верху

при помощи align="top"</p>
<p>Значение "middle" атрибута align

центрирует изображение по вертикали</p>
```

<p>Выравнивание изображения по нижнему краю  
  
(align="bottom") используется по умолчанию (можно сравнить с предыдущим примером).</p>

<p>В этом примере изображение  
  
смещено влево и начиная со следующей строки текст как бы обтекает вставленный рисунок.  
При таком способе выравнивания желательно помещать изображение в начале абзаца.</p>  
<p>  
Изображение прижато к правому краю рабочей области (align="right").  
Текст занимает все свободное пространство слева от изображения обтекая его.</p>  
</body>  
</html>

Кроме того, для форматирования изображений внутри текста используются атрибуты hspace и vspace, которые определяют интервал между графическим изображением и обтекающим текстом по горизонтали и вертикали соответственно. Размер отступов задается в пикселях.

<html>  
<body>  
<p>  
В данном варианте используются установленные по умолчанию значения hspace и vspace.  
Изображение выравнено по левому краю при помощи align="left".</p>  
<p>  
В данном варианте значения hspace и vspace установлены равными 20.  
Обратите внимание, что эти параметры влияют не столько на отступ от текста до изображения, сколько на величину свободной зоны вокруг изображения вообще. В данном случае изображение сместилось вправо вниз.  
</p>  
<p>Примечание: Для лучшего понимания работы атрибутов hspace и vspace в данном примере уменьшите размер окна Вашего браузера.  
</p>  
</body>  
</html>

### Обрамление изображения

Изображение в HTML может быть обрамлено прямоугольной рамкой. Для этого используется атрибут border, который задает ширину рамки вокруг изображения в пикселях.

<html>  
<body>  
<p>  
Изображение без рамки (по умолчанию для простых графических изображений).</p>  
<p>  
В данном варианте толщина рамки установлена в 5 пикселей.  
</p>

```
</body>
</html>
```

По умолчанию для простых изображения толщина рамки устанавливается равной нулю (отображаются без рамки). В противоположность этому для изображений, являющихся гиперссылками, по умолчанию устанавливается рамка толщиной 2 пикселя.

```
<html>
<body>
<p><a href="http://www.mggtk.ru"></a>
Несмотря на то, что значение border не задано,
изображение-гиперссылка выводится обрамленным рамкой шириной 2 пикселя (значение по
умолчанию)</p>
<p><a href="http://www.mggtk.ru"></a>
В данном варианте рамка вокруг гиперссылки убрана благодаря принудительному присвоению параметру
border нулевой толщины.
</p>
</body>
</html>
```

В последнее время атрибут border при разработке Web-сайтов в подавляющем большинстве случаев используется именно для отмены рамки вокруг изображения-гиперссылки.

### Изменение размера изображения

По умолчанию изображение отображается в браузере с сохранением своих исходных размеров и пропорций. При этом, если изображение превышает размер окна браузера, то оно может либо автоматически уменьшиться до размера окна, либо предоставить возможность просмотра по частям с использованием полос прокрутки.

При помощи атрибутов height и width (высота и ширина, соответственно) можно задать размеры отображения графического файла на Web-странице.

```
<html>
<body>
В первом случае изображение выводится с сохранением своих исходных размеров
<br>

<br>
```

Во втором варианте высота изображения уменьшена до 100 пикселей.

Обратите внимание, хотя явно задан только атрибут height, браузер автоматически отмасштабировал изображение и по ширине с сохранением исходных пропорций.

```
<br>

<br>
```

В третьем варианте принудительно заданы оба атрибута, при этом высота уменьшена до 100 пикселей, а ширина оставлена на прежнем уровне.

В этом случае автоматического масштабирования не производится, и изображение сжимается по вертикали.

```
<br>

</body>
</html>
```

Автоматическое масштабирование с заданием только одного из атрибутов довольно часто используется при необходимости вывода изображений различного формата в столбец (при этом для всех изображений задается одинаковым атрибут width) или в строку друг за другом (height).

Значение атрибутов height и width задается не только в пикселях, но и в процентах. Но в этом случае речь идет о выделенной под размещение изображения области. Это может быть все окно браузера или ячейка таблицы.

```
<html>
<body>
Ширина изображения в данном примере задана равной 100%.
Изменяя размер окна браузера можно заметить, как изображение автоматически масштабируется,
занимая всю ширину окна.
<br>

<br>
</body>
</html>
```

Необходимо отметить, что при задании значений атрибутов height и width в процентах используется тот же принцип масштабируемости, что и для значений в пикселях. При задании одного атрибута значение второго автоматически пересчитывается исходя из сохранения пропорции. При задании значений обоих атрибутов условие сохранения пропорций не соблюдается.

При размещении изображений на Web-странице желательно придерживаться следующих рекомендаций:

- При размещении изображения на Web-странице независимо от исходного размера изображения желательно в явном виде задавать размеры области для отображения при помощи атрибутов height и width. В этом случае браузер еще до полной загрузки изображения получает информацию о его размерах и заранее резервирует место, благодаря чему весь текст на странице может быть размещен и отформатирован еще до загрузки изображений.
- Желательно избегать предлагаемых браузерами возможностей масштабирования изображения (за исключением изображений заднего плана) и отображать полноразмерную графику. Для этого изображения должны быть соответствующим образом обработаны при помощи специальных графических пакетов (например, Adobe Photoshop). Это позволяет избежать искажений изображения вследствие масштабирования, уменьшает объем загружаемой информации при использовании изображения большего размера.
- При необходимости размещения на Web-странице изображений большого размера (фотографии, сложные схемы, репродукции картин и т.п.) на первой странице желательно размещать уменьшенные копии изображений, при нажатии на которые будет показываться полноразмерное изображение. В этом случае необходимо использовать два различных варианта этого изображения: полноразмерное и изображение-иконку.

Можно пояснить эти рекомендации на следующем примере.

```
<html>
```

```
<body>
```

В первом варианте в качестве изображения-иконки используется большое изображение, размер которого изменен при помощи атрибутов width и height.

При нажатии на него исходная фотография будет отображена в полный размер.

```
<br>
```

```
<a href="photo_big.png"></a>
```

```
<br>
```

Во втором варианте в качестве изображения-иконки используется специально подготовленное малое изображение, значения атрибутов width и height заданы для обеспечения правильного форматирования Web-страницы на этапе частичной загрузки.

```
<br>
```

```
<a href="photo_big.png"></a>
```

```
<br>
```

Несмотря на то, что оба этих варианта с внешней точки зрения практически идентичны, во втором варианте объем загружаемой информации значительно ниже. Так объем большого изображения "photo\_big.png" составляет более 1 Мб, в то время как "photo\_small.png" - только 14 Кб.

```
<br>
```

Очевидно, что при малой скорости подключения в Интернет пользователь может просто не дожидаться конца загрузки страницы.

```
</body>
```

```
</html>
```

### Атрибут Alt

В ряде случаев графическое изображение на Web-странице не может быть отражено. Это может происходить при просмотре страниц с использованием текстовых браузеров, при запрете на отображение графики в целях увеличения скорости загрузки и снижения расходов на Интернет, а также в ряде других случаев. При этом смысл страницы может быть существенно искажен, а при использовании графических меню и кнопок навигация по сайту будет значительно затруднена. Во избежание этого в HTML используется атрибут alt, который задает цепочку символов (максимальная длина 1024 символа), отображаемую в браузере вместо изображения и в произвольной форме описывающую его (заменяющий текст).

Следующий пример демонстрирует синтаксис применения атрибута alt.

```
<html>
```

```
<body>
```

```

```

```
<br>
```

Для демонстрации работы атрибута alt необходимо в свойствах обозревателя запретить использование графики на странице.

```
<br>
```

Обратите внимание, что даже при отображенной графике замещающий текст появляется на экране при наведении указателя мыши на изображение.

Благодаря этому атрибут alt можно использовать и в качестве своеобразной подписи к рисунку.

```
</body>
```

```
</html>
```

Использование атрибута alt считается признаком хорошего HTML-тона, так как это позволяет улучшить вывод и полезность документа для людей, который используют браузеры только для вывода текста. Кроме того, использование атрибута ALT можно считать обязательным для графических ссылок. В следующем примере приведен фрагмент графического меню. В примере специально даются ссылки на несуществующие графические файлы (отпадает необходимость запрета на отображение рисунков браузером для понимания работы примера).

```
<html>
```

```
<body>
```

В первом варианте атрибут alt не используется, пользователь видит перед собой три прямоугольника, о назначении которых он может только догадываться.

```
<hr>
```

```

```

```

```

```

```

```
<hr>
```

```
<br>
```

Использование атрибута alt во втором варианте позволяет эффективно использовать ссылки меню и в отсутствие изображений.

```
<hr>
```

```

```

```

```

```
<br>
```

```
<hr>
```

```
</body>
```

```
</html>
```