

Что такое гиперссылки. Элемент "a"

До сих пор мы с вами работали с такой единицей информации, как web-страница. Сегодня мы введем понятие **web-сайта**. Говоря просто, web-сайт - это совокупность связанных между собой страниц, объединенных общей идеей. Связать страницы между собой позволяют **гиперссылки**.

Гиперссылка - это активный элемент страницы, позволяющий *ссылаться* (указывать) на другие html страницы и прочие ресурсы в сети. При щелчке на гиперссылке в браузер загружается указанный ресурс.

Элемент "a"

Гиперссылки создаются с помощью элемента `<a>`. Помимо общих атрибутов, элемент **a** может иметь следующие атрибуты:

- **href** - адрес ресурса, на который необходимо сослаться.
- **target** - определяет, окно браузера, в которое должен быть загружен ресурс. Чаще всего используется совместно с *фреймами* (изучаются в одном из последующих занятий). Может принимать следующие значения:
 - `_blank` - ресурс загружается в новое окно;
 - `_parent` - ресурс загружается в родительском окне текущего документа;
 - `_top` - ресурс загружается в окне верхнего уровня фреймовой структуры;
 - `_self` - ресурс загружается в текущем окне. Является значением по умолчанию;
 - *"имя окна"* - ресурс загружается в заданном окне (фрейме).
- **tabindex** - число, определяющее последовательность активизации элемента кнопкой "Tab".
- **accesskey** - символ, используемый как комбинация клавиш в сочетании с клавишей **Alt**. Нажатие сочетания активизирует ссылку. Этот атрибут не получил широкого применения.

Существуют еще ряд атрибутов, не получивших особого распространения. О них вы можете узнать в справочниках или на сайте [W3C](#).

Способы адрессации

Уделим особое внимание способу указания адреса ресурса. Некоторые типичные случаи перечислены ниже:

- **Абсолютная локальная ссылка.** Указывает адрес ресурса на текущем компьютере в формате

`file://диск:\папка\...\папка\файл`

- **Абсолютная сетевая ссылка.** Указывает адрес ресурса на соседнем компьютере в сети в формате

`file://\хост\папка\...\папка\файл`

Здесь "хост" - имя компьютера или его IP адрес.

- **HTTP ссылка.** Указывает адрес ресурса, доступного по протоколу HTTP (Протокол Передачи Гипертекста). Используется следующий формат:

`http://хост:порт/папка/.../папка/файл`

Компонент "порт", впрочем, не является обязательным и используется в основном для соединения по защищенному каналу, или когда HTTP сервер работает не на порту по умолчанию (по умолчанию 80).

- **FTP ссылка.** Ссылка на ресурс, доступный по протоколу FTP (Протокол передачи файлов). Формат:

`ftp://пользователь:пароль@хост:порт/папка/.../папка/файл`

Имя пользователя и пароль используются для входа по учетной записи. Когда необходимо войти на сервер анонимно (если разрешен анонимный доступ к FTP серверу), их указывать не требуется. Компонент "порт" так же не является обязательным.

- **Почтовая гиперссылка.** Запускает диалоговое окно системного почтового клиента для создания нового сообщения электронной почты. Клиентом по умолчанию для MS Windows является MS Outlook/ Формат почтовой гиперссылки:

`mailto:пользователь@хост`

- **Относительная ссылка.** Используется для указания местоположения ресурса относительно текущего каталога. Применяются следующие обозначения:

- `/` - корневой каталог web-сервера или логического диска
- `./` - текущий каталог
- `../` - родительский каталог
- **каталог/** - дочерний каталог

В частном случае, когда адресуемый ресурс находится в одном каталоге с документом, в качестве адреса достаточно указать имя файла. Относительные ссылки обычно используют для указания адресов "внутренних" ресурсов сайта. Такой подход позволяет сделать перенос сайта быстрым и легким процессом, без последствий в виде перенастройки ссылок.

Мы рассмотрим этот способ адрессации более подробно на примере построения фотогалереи в конце урока.

Практический пример

Итак, попробуем проверить теорию практикой. Пусть необходимо разметить в XHTML документ, разбитый на 5 разделов. Каждый раздел - отдельный html-файл. Содержание (набор ссылок) создано в отдельном файле - "index.html". Нас не особо интересует содержимое самих разделов. Особое внимание мы уделим созданию ссылок на них из содержания.

```

<h1>Дэйв Реггет: Основы HTML</h1>
<h2>Содержание</h2>
<ul>
  <li><a href="chapter1.html">Начните с заголовка</a></li>
  <li><a href="chapter2.html">Добавление в страницу
изображений</a></li>
  <li><a href="chapter3.html">Три вида списков</a></li>
  <li><a href="chapter4.html">Дополнительная
информация</a></li>
</ul>

<p style="font-style: italic; font-size: 10pt; color:
#666666;">Это
руководство по <b>HTML</b>, поэтому некоторые правила не
вписываются,
а то и нарушают ту картину, которую рисуем вам мы с начала курса
XHTML.
Автором руководства является
<a href="http://www.w3.org/People/Raggett"
target="_blank">Дэйв Реггет</a>. Более полную информацию по HTML
и XHTML
вы найдете на сайте
<a href="http://www.w3.org" target="_blank">W3C</a>.
</p>

```

Обратите внимание, что ссылки в нижней части документа (последний абзац) открываются в новом окне, благодаря наличию у них атрибута `target="_blank"`. Так обычно поступают с теми ссылками, которые указывают на внешние по отношению к вашему сайту ресурсы. Таким образом, пользователь не "потеряет" ваш сайт, перейдя по ссылке.

Графические ссылки

В элемент **a** можно вкладывать не только текст, но и изображение. Именно так и создаются графические ссылки. Графические ссылки в ряде случаев выглядят гораздо эстетичнее чем текстовые. Здесь, однако, нас подстерегает не очень приятный сюрприз - изображения, заключенные в элемент **a** приобретают рамку. Для того, чтобы избавиться от нее необходимо использовать атрибут **border** элемента **img**:

```

<a href="..."></a>

```

Еще раз напоминаю вам, что при использовании изображений в качестве ссылок, не забудьте указать для них атрибут **alt**. В случае невозможности отображения рисунка, браузер сделает гиперссылкой его текстовое описание.

Пример: создание навигационной панели сайта

Сейчас на небольшом примере мы с вами поэтапно спроектируем и реализуем несложную графическую навигационную панель сайта. Выглядеть это будет так:



Кажется, что все просто: берем изображение, размещаем его на странице... Что может быть проще? Однако, различные фрагменты этого изображения должны быть гиперссылками на различные ресурсы. Таким образом, мы приходим к необходимости разрезать рисунок на несколько частей.

Вот что получилось у меня:



В результате, у нас есть три "кнопки" - гиперссылки "Назад", "Содержание" и "Вперед". Разместим все это добро на странице:

```
<div style="text-align: center;">
  
  
  ...
  
</div>
```

Как говорится - "Первый блин комом"! Между изображениями появились зазоры, хотя никто этого не планировал. Все дело в том, что MSIE добавил между изображениями пробельный символ. Это происходит потому, что элементы **img** разделены переводом строки в коде страницы. Зная это, положение вещей легко помянуть, удалив пробельные символы (переводы строки, табуляции...) между элементами **img**.

```

```

Теперь все в норме! Снабдим код гиперссылками. Чтобы не придумывать адреса, в качестве значения атрибута **href** используйте "#":

Полученный результат выглядит явно как то не так... Все дело в рамках вокруг изображений. Избавимся от них, добавив в изображения-ссылки атрибут **border="0"**:

```
<a href="#"></a>...
```

Теперь вы можете связать несколько страниц несложной единой навигационной панелью, разместив ее код в начале **каждой** страницы и правильно настроив ссылки.

