

12 Ссылки

12.1 Введение в ссылки и якоря

HTML предлагает множество условных оборотов для текстовых и структурированных документов, но что отличает его от большинства других языков разметки - его возможности разметки гипертекста и интерактивных документов. В этом разделе вводится понятие *ссылки* (или гиперссылки или ссылки Web), основной гипертекстовой конструкции. Ссылки связывают один ресурс Web с другим. Несмотря на простоту, ссылки стали основным залогом успеха Web.

Ссылка имеет два конца - называемых *anchors* -- и направление. Ссылка начинается в "исходном" anchor (источнике) и указывает на "целевой" anchor, который может быть любым ресурсом Web (например, изображением, видеоклипком, звуковым файлом, программой, документом HTML, элементом в документе HTML и т.д.).

12.1.1 Переход к ресурсу, на который указывает ссылка

По умолчанию со ссылкой связана загрузка другого ресурса Web. Это поведение достигается обычно путем выбора ссылки (например, с помощью щелчка мыши, ввода с клавиатуры и т.д.).

В следующем HTML-фрагменте содержится две ссылки, у одной целевым anchor является документ HTML с именем "chapter2.html", а у второй целевой anchor - изображение в формате GIF, расположенное в файле "forest.gif":

```
<BODY>
...какой-то текст...
<P>Подробнее см. в <A href="chapter2.html">главе два</A>.
См. также <A href="../images/forest.gif">карту леса.</A>
</BODY>
```

Путем активизации этих ссылок (с помощью щелчка мыши, ввода с клавиатуры, голосовых команд и т.д.) пользователи могут перейти к этим ресурсам. Обратите внимание, что атрибут `href` в каждом исходном anchor указывает адрес целевого anchor с использованием URI. Целевой anchor ссылки может быть элементом в документе HTML. Целевому anchor должно даваться имя и адрес URI, адресующий этот anchor. Должен содержать это имя в качестве *идентификатора фрагмента*. Целевые anchors в документах HTML могут указываться с помощью элемента `A` (именующего его с помощью атрибута `name`) или с помощью любого другого элемента (именующего с помощью атрибута `id`).

Таким образом, например, автор может создавать оглавление, элементы которого являются ссылками на элементы заголовков `H2`, `H3` и т.д. в том же документе. Используя элемент `A` для создания целевых anchors, можно записать:

```
<H1>Содержание;/H1>
<P><A href="#section1">Введение</A><BR>
<A href="#section2">Предыстория</A><BR>
<A href="#section2.1">Более конкретные заметки</A><BR>
...продолжение содержания...
...тело документа...
<H2><A name="section1">Введение</A></H2>
...раздел 1...
<H2><A name="section2">Предыстория</A></H2>
...раздел 2...
<H3><A name="section2.1">Более конкретные заметки</A></H3>
...раздел 2.1...
```

Этого же эффекта можно достигнуть, сделав якорями сами элементы заголовка:

```
<H1>Содержание</H1>
<P><A href="#section1">Введение</A><BR>
<A href="#section2">Предыстория</A><BR>
<A href="#section2.1">Более конкретные заметки</A><BR>
...продолжение содержания...
...тело документа...
```

```
<H2 id="section1">Введение</H2>
... раздел 1...
<H2 id="section2">Предыстория</H2>
... раздел 2...
<H3 id="section2.1">Более конкретные заметки</H3>
...раздел 2.1...
```

12.1.2 Другие отношения ссылок

Наиболее часто сейчас ссылки *используются* для загрузки других ресурсов Web, как показано в предыдущих примерах. Однако авторы могут помещать в свои документы ссылки, выражающие иные отношения между ресурсами, чем просто "использование ссылки для Просмотра связанного ресурса". Для ссылок, выражающих другие типы отношений, В исходном якоре указываются один или более [ТИПОВ ССЫЛОК](#).

Роли ссылок, определяемых элементом [A](#) или [LINK](#) указываются с помощью атрибутов [rel](#) и [rev](#).

Например, ссылки, определяемые элементом [LINK](#), могут описывать положение документа в последовательности документов. В следующем примере ссылка в документе "Глава 5" указывает на предыдущую и последующую главы:

```
<HEAD>
...другая заголовочная информация...
<TITLE>Глава 5</TITLE>
<LINK rel="prev" href="chapter4.html">
<LINK rel="next" href="chapter6.html">
</HEAD>
```

Тип первой ссылки - "prev", второй - "next" (два из ряда распознаваемых [ТИПОВ ССЫЛОК](#)). Ссылки, задаваемые элементов [LINK](#), **не** генерируются с содержимым документа, хотя агенты пользователей могут отображать их различными способами (например, в виде средств перехода).

Даже если они не используются для перехода, эти ссылки могут интерпретироваться интересными способами. Например, агент пользователя, выполняющий печать ряда документов в формате HTML в виде одного документа, может использовать эту информацию для формирования последовательного линейного документа. Дальнейшая информация об использовании [ССЫЛОК ДЛЯ ПОИСКОВЫХ СИСТЕМ](#) приведена ниже.

12.1.3 Задание якорей и ссылок

Хотя некоторые элементы и атрибуты языка HTML создают ссылки на другие ресурсы (например, элемент [IMG](#), элемент [FORM](#) и т.д.), в этой главе обсуждаются ссылки и якоря, создаваемые элементами [LINK](#) и [A](#). Элемент [LINK](#) может присутствовать только в заголовке документа. Элемент [A](#) может присутствовать только в теле документа. Если для элемента [A](#) установлен атрибут [href](#), этот элемент определяет исходный якорь для ссылки, которая может активизироваться пользователем для загрузки ресурса Web. Исходный якорь - это местоположение экземпляра элемента [A](#) и целевой якорь ресурса Web.

Загруженный ресурс может обрабатываться агентом пользователя различными способами: открытием нового документа HTML в том же окне агента, открытием нового документа HTML в другом окне, запуском новой программы для обработки ресурса и т.д. Поскольку элемент [A](#) имеет содержание (текст, изображения и т.д.), агенты пользователей могут генерировать это содержимое так, чтобы показать наличие ссылки (например, подчеркивая содержимое). Если установлены атрибуты [name](#) или [id](#) элемента [A](#), этот элемент определяет якорь, который может служить целью других ссылок. Авторы могут устанавливать атрибуты [name](#) и [href](#) одновременно в одном экземпляре элемента [A](#). Элемент [LINK](#) определяет отношение между текущим документом и другим ресурсом. Хотя элемент [LINK](#) не имеет содержимого, определяемые им отношения могут отображаться некоторыми агентами пользователей.

12.1.4 Заголовки ссылок

Атрибут `title` может устанавливаться как для элемента `A`, так и для элемента `LINK`; в нем указывается информация о природе ссылки. Эта информация может проговариваться агентом пользователя, отображаться в виде подсказки, изменять вид курсора и т.д.

Таким образом можно `augment` [предыдущий пример](#), указав для каждой ссылки заголовок:

```
<BODY>
...некоторый текст...
<P>Подробнее Вы можете узнать об этом в <A href="chapter2.html"
    title="Перейдите к главе 2">главе 2</A>.
<A href="/chapter2.html"
    title="Переход к главе 2.">глава 2</A>.
См. также <A href="../images/forest.gif"
    title="Изображение леса в формате GIF">карту леса.</A>
</BODY>
```

12.1.5 Интернационализация и ссылки

Поскольку ссылки могут указывать на документы, использующие другие [кодировки символов](#), элементы `A` и `LINK` поддерживают атрибут `charset`. Этот атрибут позволяет авторам рекомендовать агентам пользователей кодировку данных в связанном документе. Атрибут `hreflang` предоставляет агенту пользователя информацию о языке связанного ресурса, точно так же, как атрибут `lang` предоставляет информацию о языке содержимого элемента или значениях атрибутов.

Имея дополнительные знания, агенты пользователей должны иметь возможность не представлять пользователю "мусор". Вместо этого они могут находить ресурсы, необходимые для корректного представления документа или, если они не могут найти эти ресурсы, по крайней мере уведомить пользователя о невозможности прочтения документа с указанием причины.

12.2 Элемент A

```
<!ELEMENT A - - (%inline;)* -(A) -- якорь (точка ссылки) -->
<!ATTLIST A
  %attrs; -- %coreattrs, %i18n, %events --
  charset %Charset; #IMPLIED -- кодировка символов связанного ресурса --
  type %ContentType; #IMPLIED -- рекомендуемый тип содержимого --
  name CDATA #IMPLIED -- конец именованной ссылки --
  href %URI; #IMPLIED -- URI связанного ресурса --
  hreflang %LanguageCode; #IMPLIED -- код языка --
  rel %LinkTypes; #IMPLIED -- типы прямых ссылок --
  rev %LinkTypes; #IMPLIED -- типы обратных ссылок --
  accesskey %Character; #IMPLIED -- клавиша доступа --
  shape %Shape; rect -- для использования с клиентскими навигацион-
ными картами --
  coords %Coords; #IMPLIED -- для использования с клиентскими навигацион-
ными картами --
  tabindex NUMBER #IMPLIED -- положение в последовательности перехода --
  onfocus %Script; #IMPLIED -- фокус на элементе --
  onblur %Script; #IMPLIED -- фокус вне элемента --
  >
```

Начальный тэг: **обязателен**, Конечный тэг: **обязателен**

Определения атрибутов

`name` = `CDATA` [CS]

Этот атрибут дает якорю имя, так что он может служить целью другой ссылки. Значением этого атрибута должно быть уникальное имя. Область действия имени - текущий документ. Обратите внимание, что этот атрибут использует общее пространство имен с атрибутом `id`.

`href` = `uri` [CT]

Этот атрибут определяет местоположение ресурса Web, определяя связь между текущим элементом (исходным якорем) и целевым якорем, определяемым этим атрибутом.

hreflang = [langcode](#) [CI]

Этот атрибут задает базовый язык ресурса, назначаемого атрибутом [href](#), и может использоваться, только если указан атрибут [href](#).

type = [content-type](#) [CI]

Если этот атрибут задан, он определяет тип части содержимого, например, результат обратного действия ссылки на ресурс. Типы содержимого определены в разделе [\[MIMETYPES\]](#).

rel = [link-types](#) [CI]

Этот атрибут описывает отношение из текущего документа к якорю, заданному атрибутом [href](#). Значением этого атрибута является разделенный пробелами список типов ссылок.

rev = [link-types](#) [CI]

Этот атрибут используется для описания [обратной ссылки](#) из якоря, задаваемого атрибутом [href](#), в текущий документ. Значением этого атрибута является разделенный пробелами список типов ссылок.

charset = [charset](#) [CI]

Этот атрибут задает кодировку символов назначаемого ссылкой ресурса. Подробнее см. в разделе о [кодировках символов](#).

Атрибуты, определяемые в другом месте

- [id](#), [class](#) (идентификаторы в пределах документа)
- [lang](#) (информация о языке), [dir](#) (направление текста)
- [title](#) (заголовок элемента)
- [style](#) (встроенная информация о стиле)
- [shape](#) and [coords](#) (навигационные карты)
- [onfocus](#), [onblur](#), [onclick](#), [ondblclick](#), [onmousedown](#), [onmouseup](#), [onmouseover](#), [onmousemove](#), [onmouseout](#), [onkeypress](#), [onkeydown](#), [onkeyup](#) (внутренние события)
- [target](#) (информация о целевом кадре)
- [tabindex](#) (последовательность перехода)
- [accesskey](#) (клавиши доступа)

Каждый элемент [A](#) определяет якорь

1. Содержимое элемента [A](#) определяет положение якоря.
2. Атрибут [name](#) задает имя якоря, так что он может служить пунктом назначения любого числа ссылок (см. также [якоря с атрибутом id](#)).
3. Атрибут [href](#) назначает якорь пунктом назначения ровно одной ссылки.

Авторы также могут создавать элемент [A](#), не определяющий ни одного якоря, и не указывающий [href](#), [name](#) или [id](#). Значения этих атрибутов могут устанавливаться позднее с помощью [скриптов](#).

В следующем примере элемент [A](#) определяет ссылку. Исходным якорем является текст " Web-сайте W3C", а пунктом назначения - "http://www.w3.org/":

```
Подробнее о W3C Вы можете узнать на  
<A href="http://www.w3.org/"> Web-сайте W3C</A>.
```

Эта ссылка указывает на страницу World Wide Web Consortium. Если пользователь активизирует эту ссылку в своем агенте, агент загрузит ресурс, в данном случае - документ HTML.

Обычно агенты пользователей представляют ссылки так, чтобы они были очевидны для пользователей (подчеркивание, обратное видео и т.д.). Конкретное представление зависит от агента пользователя. Представление может различаться в зависимости от того, посетил ли уже пользователь указанную ссылку. Возможное визуальное представлением предыдущей ссылки:

```
Подробнее о W3C Вы можете узнать на Web-сайте W3C.
```

~~~~~

Чтобы дать агентам пользователей явное указание кодировки символов целевой страницы, установите атрибут `charset`:

```
Подробнее о W3C Вы можете узнать на
<A href="http://www.w3.org/" charset="ISO-8859-1">Web-сайте W3C</A>
```

Предположим, мы определяем якорь с именем "anchor-one" в файле "one.html".

```
...текст до якоря...
<A name="anchor-one">Это местоположение первого якоря.</A>
...текст после якоря...
```

В данном фрагменте вокруг текста "Это местоположение первого якоря" создается якорь. Обычно содержимое элемента `A` не представляется каким-либо особым образом, если элемент `A` определяет только якорь.

Определив якорь, мы можем связать его с этим же или с другим документом. URI, назначающие якорь, включают символ "#", за которым следует имя якоря ([идентификатор фрагмента](#)). Вот несколько примеров таких URI:

- Абсолютный URI: `http://www.mycompany.com/one.html#anchor-one`
- Относительный URI: `./one.html#anchor-one` или `one.html#anchor-one`
- Если ссылка определена в том же документе: `#anchor-one`

Таким образом, ссылка, определенная в файле "two.html", который находится в одном каталоге с файлом "one.html", будет указывать на этот якорь следующим образом:

```
...текст до ссылки...
Подробнее см. <A href="./one.html#anchor-one"> первый якорь</A>.
...текст после ссылки...
```

Элемент `A` в следующем примере задает ссылку (с помощью атрибута `href`) и одновременно создает именованный якорь (с помощью атрибута `name`):

```
Я только что вернулся из отпуска! Вот
<A name="anchor-two"
  href="http://www.somecompany.com/People/Ian/vacation/family.png">
фотография моей семьи на озере.</A>.
```

В этом примере содержится ссылка на другой тип ресурса Web (изображение в формате PNG). Активизация ссылки должна привести к загрузке изображения из Web (и, возможно, его отображению, если система сконфигурирована соответствующим образом).

***Примечание.** Агенты пользователей должны уметь находить якоря, созданные пустыми элементами `A`, но некоторые агенты не могут этого делать. Например, некоторые агенты пользователей могут не найти "empty-anchor" в следующем фрагменте HTML:*

```
<A name="empty-anchor"></A>
<EM>...некоторые текст в формате HTML...</EM>
<A href="#empty-anchor">Ссылка на пустой якорь</A>
```

## 12.2.1 Синтаксис имен якорей

Именем якоря является значение атрибута `name` или атрибута `id`, используемого в контексте якоря. К именам якорей применяются следующие правила:

- **Уникальность:** Имена якорей должны быть уникальны в пределах документа. Имена якорей в одном документе не могут отличаться только регистром.
- **Соответствие строк:** Сравнение между [идентификаторами фрагментов](#) и именами якорей должно проводиться на основе полного (с учетом регистра) совпадения. Таким образом, следующий пример является корректным в смысле соответствия строк и должен рассматриваться агентами пользователей как совпадение:

```
<P><A href="#xxx">...</A>
...некоторый текст...
```

```
<P><A name="xxx">...</A>
```

#### ПРИМЕР НЕДОПУСТИМОГО ИСПОЛЬЗОВАНИЯ:

Следующий пример не является корректным в смысле уникальности, поскольку два имени отличаются только регистром:

```
<P><A name="xxx">...</A>
<P><A name="XXX">...</A>
```

Хотя далее приводится допустимый код HTML, поведение агента пользователя в данном случае не определено; некоторые агенты пользователей могут (ошибочно) считать это совпадением, другие могут так не считать.

```
<P><A href="#xxx">...</A>
...некоторый текст...
<P><A name="XXX">...</A>
```

Имена якорей должны содержать только символы набора ASCII. Подробнее см. в разделе [об использовании символов, не входящих в набор ASCII, в значениях атрибутов URI](#).

### 12.2.2 Вложенные ссылки недопустимы

Ссылки и якоря, определяемые элементом [A](#), не могут быть вложенными; элемент [A](#) не должен содержать других элементов [A](#).

Поскольку DTD определяет элемент [LINK](#) как пустой, элементы [LINK](#) также не могут быть вложенными.

### 12.2.3 Якоря с атрибутом id

Атрибут [id](#) может использоваться для создания якоря в начальном тэге любого элемента (включая элемент [A](#)).

В данном примере показано использование атрибута [id](#) для размещения якоря в элементе [H2](#). Связь с якорем осуществляется с помощью элемента [A](#).

```
Подробнее об этом Вы можете прочесть в <A href="#section2">Разделе 2</A>.
...далее в документе
<H2 id="section2">Раздел 2</H2>
...далее в документе
<P>Подробнее см. в <A href="#section2">Разделе 2</A> выше.
```

В следующем примере имя целевого якоря задается с помощью атрибута [id](#):

```
Я только что вернулся из отпуска! Вот
<A id="anchor-two">фотография моей семьи на озере.</A>.
```

Атрибуты [id](#) и [name](#) используют одно и то же пространство имен. Это означает, что они не могут определять якорь с одним и тем же именем в одном и том же документе.

#### ПРИМЕР НЕДОПУСТИМОГО ИСПОЛЬЗОВАНИЯ:

Далее показан недопустимый фрагмент кода HTML, поскольку эти атрибут объявляют одно и то же имя дважды в пределах одного документа.

```
<A href="#a1">...</A>
...
<H1 id="a1">
...страницы и страницы...
<A name="a1"></A>
```

По спецификации в HTML DTD атрибут [name](#) может содержать [ссылки на символы](#). Таким образом, значение `D&#xfc;rst` является допустимым для атрибута [name](#), так же как `D&uuml;rst`. Атрибут [id](#) не может содержать ссылки на символы.

**Что использовать - [id](#) или [name](#)?** При выборе элемента [id](#) или [name](#) для указания имени якоря авторы должны иметь в виду следующее:

- Атрибут [id](#) может быть не только именем якоря (а также селектором таблицы стилей, идентификатором обработки и т.д.).



- *Некоторые агенты пользователей более старых версий не поддерживают якоря, созданные с помощью атрибута [id](#).*
- *Атрибут [name](#) обеспечивает более разнообразные имена якорей (с использованием [entities](#)).*

## 12.2.4 Недоступные и неопределяемые ресурсы

Ссылка на недоступный или неопределяемый ресурс является ошибкой. Хотя агенты пользователей могут по-разному обрабатывать такие ошибки, рекомендуется следующее:

- Если агент пользователя не может найти связанный ресурс, он должен уведомить об этом пользователя.
- Если агент пользователя не может идентифицировать тип связанного ресурса, он должен попытаться обработать его. Он должен уведомить пользователя и позволить пользователю вмешаться и определить тип документа.

## 12.3 Отношения документов: элемент LINK

```

<!ELEMENT LINK - O EMPTY          -- независимая от устройства ссылка -->
<!ATTLIST LINK
  %attrs;                          -- %coreattrs, %i18n, %events --
  charset      %Charset;           #IMPLIED -- кодировка символов связанного ресурса --
  href         %URI;               #IMPLIED -- URI связанного ресурса --
  hreflang    %LanguageCode;      #IMPLIED -- код языка --
  type        %ContentType;       #IMPLIED -- рекомендуемый тип содержимого --
  rel         %LinkTypes;         #IMPLIED -- тип прямой связи --
  rev         %LinkTypes;         #IMPLIED -- тип обратной связи --
  media       %MediaDesc;         #IMPLIED -- для представления на этих устройствах --
  >

```

*Начальный тэг: **обязателен**, Конечный тэг: **запрещен***

*Атрибуты, определяемые в другом месте*

- [id](#), [class](#) (идентификаторы в пределах документа)
- [lang](#) (информация о языке), [dir](#) (направление текста)
- [title](#) (заголовок элемента)
- [style](#) (встроенная информация о стиле)
- [onclick](#), [ondblclick](#), [onmousedown](#), [onmouseup](#), [onmouseover](#), [onmousemove](#), [onmouseout](#), [onkeypress](#), [onkeydown](#), [onkeyup](#) (внутренние события)
- [href](#), [hreflang](#), [type](#), [rel](#), [rev](#) (ссылки и якоря)
- [target](#) (информация о целевом фрейме)
- [media](#) (информация о стиле заголовка)
- [charset](#) (кодировки символов)

Этот элемент определяет связь. В отличие от элемента [A](#), он может присутствовать только в разделе [HEAD](#) документа, хотя может присутствовать неограниченное число раз. Хотя элемент [LINK](#) не имеет содержимого, он содержит информацию об отношениях, которая может представляться агентами пользователей различными способами (например, в виде панели с выпадающим списком ссылок).

В данном примере показано, как несколько определений элемента [LINK](#) могут быть представлены в разделе [HEAD](#) документа. Текущим документом является "Chapter2.html". Атрибут [rel](#) указывает отношение связанного документа с текущим документом. Значения "Index", "Next" и "Prev" описаны в разделе, посвященном [типам ссылок](#).

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
  "http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
  <TITLE>Глава 2</TITLE>
  <LINK rel="Index" href="../index.html">
  <LINK rel="Next" href="Chapter3.html">
  <LINK rel="Prev" href="Chapter1.html">

```

```
</HEAD>
...продолжение документа...
```

### 12.3.1 Прямые и обратные ссылки

Атрибуты [rel](#) и [rev](#) играют дополнительные роли - атрибут [rel](#) задает прямую ссылку, а атрибут [rev](#) - обратную.

Рассмотрим два документа - А и В.

```
Документ А:      <LINK href="docB" rel="foo">
```

имеет точно то же значение, что и:

```
Документ В:      <LINK href="docA" rev="foo">
```

Оба атрибута могут определяться одновременно.

### 12.3.2 Ссылки и внешние таблицы стилей

Если элемент [LINK](#) ссылается на внешнюю таблицу стилей для документа, атрибут [type](#) указывает язык таблицы стилей, а атрибут [media](#) - предполагаемое устройство или устройства для представления документа. Агенты пользователей могут сэкономить время, загружая из сети только те таблицы стилей, которые применяются к текущему устройству. [Типы устройств](#) обсуждаются в разделе, посвященном таблицам стилей.

### 12.3.3 Ссылки и поисковые машины

Авторы могут использовать элемент [LINK](#) для указания различной информации для поисковых машин, включая:

- Ссылки на альтернативные версии документа, написанные на других языках.
- Ссылки на альтернативные версии документа, разработанные для других устройств, например, на версии, предназначенные специально для печати.
- Ссылки на начальные страницы набора документов.

В примерах ниже показано, как информация о языке, типах устройств и типах ссылок может использоваться для улучшения обработки документа поисковыми машинами.

В следующем примере мы используем атрибут [hreflang](#), чтобы сообщить поисковым машинам, где находятся голландская, португальская и арабская версии документа. Обратите внимание на использование атрибутов [dir](#) и [charset](#) для руководства на арабском языке, а также на использование атрибута [lang](#) для указания того, что значением атрибута [title](#) для элемента [LINK](#), указывающего руководство на французском языке, задано по-французски.

```
<HEAD>
<TITLE>Руководство на английском языке</TITLE>
<LINK title="Руководство на голландском языке"
      type="text/html"
      rel="alternate"
      hreflang="nl"
      href="http://someplace.com/manual/dutch.html">
<LINK title="Руководство на португальском языке"
      type="text/html"
      rel="alternate"
      hreflang="pt"
      href="http://someplace.com/manual/portuguese.html">
<LINK title="Руководство на арабском языке"
      dir="rtl"
      type="text/html"
      rel="alternate"
      charset="ISO-8859-6"
      hreflang="ar"
      href="http://someplace.com/manual/arabic.html">
<LINK lang="fr" title="La documentation en Fran&ccedil;ais"
      type="text/html"
      rel="alternate"
      hreflang="fr"
      href="http://someplace.com/manual/french.html">
```



```
</HEAD>
```

В следующем примере мы сообщаем поисковым машинам, где находится печатная версия.

```
<HEAD>
<TITLE>Руководство</TITLE>
<LINK media="print" title="Руководство в формате postscript"
      type="application/postscript"
      rel="alternate"
      href="http://someplace.com/manual/postscript.ps">
</HEAD>
```

В следующем примере сообщаем поисковым машинам, где находится первая страница.

```
<HEAD>
<TITLE>Руководство - страница 5</TITLE>
<LINK rel="Start" title="Первая страница руководства"
      type="text/html"
      href="http://someplace.com/manual/start.html">
</HEAD>
```

Дальнейшая информация приведена в замечаниях в приложении о том, [как помочь поисковым машинам проиндексировать Ваш Web-сайт](#).

## 12.4 Информация о пути: элемент BASE

```
<!ELEMENT BASE - O EMPTY -- базовый URI документа -->
<!ATTLIST BASE
  href %URI; #REQUIRED -- URI, служащий базовым --
>
```

*Начальный тэг: **обязателен**, Конечный тэг: **запрещен***

*Определения атрибутов*

href = [uri](#) [CT]

Этот атрибут задает абсолютный URI, служащий базовым URI для разрешения относительных URI.

*Атрибуты, определяемые в другом месте*

- [target](#) ([информация о целевом кадре](#))

В HTML ссылки на внешние изображения, апплеты, программы для обработки форм, таблицы стилей и т.д. всегда задаются с помощью URI. Относительные URI [разрешаются](#) в соответствии с базовым URI, который может определяться из различных источников. Элемент [BASE](#) позволяет авторам явно указать базовый URI документа.

Если элемент [BASE](#) указан, он должен присутствовать в разделе [HEAD](#) документа HTML, до элементов, ссылающихся на внешние ресурсы. Информация о пути, указанная в элементе [BASE](#), влияет только на URI в документе, в котором присутствует этот элемент.

Например, даны следующее объявление [BASE](#) и объявление [A](#):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Наши продукты</TITLE>
<BASE href="http://www.aviary.com/products/intro.html">
</HEAD>
<BODY>
<P>Вы видели наши <A href=" ../cages/birds.gif">клетки для птиц</A>?
</BODY>
</HTML>
```

относительный URI `../cages/birds.gif` будет разрешен в:

```
http://www.aviary.com/cages/birds.gif
```

### 12.4.1 Разрешение относительных URI

Агенты пользователей должны вычислять базовый URI для разрешения относительных URI в соответствии с [\[RFC1808\]](#), раздел 3. Далее описано, как [\[RFC1808\]](#) применяется именно к HTML.

Агенты пользователей должны вычислять базовый URI в соответствии со следующим приоритетом (от высшего приоритет к низшему):

1. Базовый URI, устанавливаемый элементом [BASE](#).
2. Базовый URI, задаваемый метаданными, обнаруженными в процессе работы по протоколу, такими как заголовок HTTP (см. [\[RFC2068\]](#)).
3. По умолчанию базовым URI является URI текущего документа. Не все документы HTML имеют базовый URI (например, документ HTML может присутствовать в сообщении электронной почты и может не определяться никаким URI). Такие документы HTML считаются *orphanous*, если они содержат относительные URI и используют базовый URI по умолчанию.

Кроме того, элементы [OBJECT](#) и [APPLET](#) определяют атрибуты, имеющие преимущество над значением, установленным для элемента [BASE](#). Подробнее об относящихся к ним определениях URI см. в определениях этих элементов.

Ссылки, указанные в заголовках HTTP, обрабатываются в точности так, как элементы [LINK](#), явно установленные в документе.