



Метаданные. Редактирование текста. Ссылки в HTML

Что такое <head>?

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="utf-8">
  <title>Тестовая
  страница</title>
</head>
<body>
  <p>Какой-то текст на этой
  странице</p>
```

Содержимое <head>, в отличие от содержимого элемента <body>, не отображается на странице. Задача <head> — хранить метаданные документа. В приведённом выше примере <head> совсем небольшой.

Название страницы (title)

Мы уже видели, как работает элемент `<title>`: его используют для добавления заголовка (названия страницы) в документ.

```
<head>  
  <meta charset="utf-8">  
  <title>Тестовая  
  страница</title>  
</head>
```

Метаданные: Элемент `<meta>`

Метаданные — данные, которые описывают данные. У HTML есть «официальное» место для метаданных документа — элемент `<meta>`. Конечно, другие вещи, о которых мы говорим в этой статье, тоже можно назвать метаданными. Существует множество разновидностей `<meta>`. Не станем пытаться охватить их все сразу — так недолго и запутаться, а рассмотрим несколько самых популярных, чтобы разобраться, что к чему.

Указываем кодировку текста документа

В этом элементе указана кодировка документа — набор символов, которые в нём можно использовать. utf-8 — универсальный набор символов, который включает почти все символы со всех языков человечества. Такая веб-страница сможет работать с любым языком. Установить эту кодировку на всех веб-страницах, которые вы создаёте — отличная идея!

```
<meta  
charset="utf-8">
```

Указываем автора и описание

У элементов <meta> часто есть атрибуты name и content:

- name — тип элемента, то есть какие именно метаданные он содержит.
- content — сами метаданные.

Два полезных элемента метаданных — указание автора страницы и краткое описание её содержимого. Рассмотрим эти элементы на примере:

```
<meta name="author" content="skr1pmen">  
<meta name="description" content="Некое  
описание проекта">
```

По указанному имени автора (author) можно найти человека, который написал страницу, и связаться с ним. Некоторые системы управления содержимым (CMS) автоматически обрабатывают эту информацию и делают её доступной для таких целей.

Viewport

```
<meta name="viewport"  
      content="width=device-width, user-scalable=no, initial-scale=1.0,  
maximum-scale=1.0, minimum-scale=1.0">
```

Метаданные viewport нужны для того чтобы в дальнейшем сайт можно было адаптировать под разные разрешения экранов.

Подключение CSS

Современные сайты используют CSS, чтобы выглядеть привлекательнее

Элемент `<link>` помещают в заголовок документа. У него есть два атрибута: `rel="stylesheet"` показывает, что мы указываем стиль документа, а в `href` указан путь к файлу:

```
<link rel="stylesheet"  
href="style.css">
```


Добавление иконок

Чтобы добавить своему сайту узнаваемости, можно указать в метаданных разные иконки.

Чтобы добавить на страницу favicon:

1. Сохраните изображение в формате .ico (многие браузеры поддерживают и в более привычных форматах, таких как .gif или .png) в папку со своим документом. Старые браузеры, например, Internet Explorer 6, поддерживают только формат .ico
2. Добавьте ссылку на иконку в <head> документа:

```
<link rel="shortcut icon" href="favicon.ico"  
type="image/x-icon">
```

Заголовки и абзацы / параграфы

Большинство структурированных текстов состоят из параграфов и заголовков, независимо от того, читаете ли вы рассказ, или газету, или учебник, журнал и т.д. Упорядоченный контент делает чтение более лёгким и приятным.

В HTML каждый абзац заключён в элемент `<p>`, подобно:

```
<p>Какой-то текст на этой  
странице</p>
```

Каждый заголовок заключён в элемент заголовка `<h1>`:

```
<h1>Заголовок первого  
уровня</h1>
```

Имеется шесть элементов заголовка: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` и `<h6>`. Каждый элемент представляет разный уровень контента в документе; `<h1>` представляет главный заголовок, `<h2>` представляет подзаголовки, `<h3>` представляет под-подзаголовки и так далее.

Списки

Списки есть везде вокруг нас — от вашего списка покупок до списка направлений, которым вы подсознательно следуете, чтобы каждый день добраться домой, и списка инструкций, которые вы выполняете в этом руководстве! Списки используются всюду в Интернете, и мы рассмотрим некоторые типа списков.

Нумерованные списки

Нумерованные списки — это списки, в которых порядок элементов имеет номер.

```
<ol>
```

```
<li>молоко</li>  
>
```

```
<li>яйца</li>
```

```
<li>хлеб</li>
```

```
<li>хумус</li>  
</ol>
```

1. молоко
2. яйца
3. хлеб
4. хумус

Акцент и важность

В обиходе мы часто подчёркиваем определённые слова, чтобы изменить смысл предложения и мы часто хотим отметить некоторые слова как важные или разные в некотором роде. HTML предоставляет различные семантические элементы, позволяющие нам добавлять текстовые материалы с такими эффектами, и в этом разделе мы рассмотрим несколько наиболее распространённых.

АКЦЕНТ

Когда мы хотим добавить акцент в разговорный язык, мы подчёркиваем определённые слова, тонко изменяя смысл того, что мы говорим. Точно так же на письменном языке мы склонны подчёркивать слова, выделяя их *курсивом*. Например, следующие два предложения имеют разные значения.

Я рад, что ты не опоздал.

Я *рад*, что ты не *опоздал*.

В первом предложении звучит искреннее облегчение, что человек не опоздал. Во втором, напротив, звучит сарказм или пассивная агрессия: так выражена досада от того, что человек немного опоздал.

Ряд элементов html предназначены для форматирования текстового содержимого, например, для выделения жирным или курсивом и т.д.

- ``: выделяет текст жирным
 - ``: зачеркивает текст
 - `<i>`: выделяет текст курсивом
- ``: выделяет текст курсивом, в отличие от тега `<i>` носит логическое значение, придает выделяемому тексту оттенок важности
 - `<s>`: зачеркивает текст
 - `<small>`: делает текст чуть меньше размером, чем окружающий
 - ``: выделяет текст жирным. В отличие от тега `` предназначен для логического выделения, чтобы показать важность текста. А `` не носит характера логического выделения, выполняет функции только форматирования
 - `<sub>`: помещает текст под строкой
 - `<sup>`: помещает текст над строкой
 - `<u>`: подчеркивает текст
 - `<ins>`: определяет вставленный (или добавленный) текст
 - `<mark>`: выделяет текст цветом, придавая ему оттенок важности

Создание гиперссылок

Гиперссылки — одно из самых интересных нововведений Интернета. Они были особенностью Сети с самого начала, но именно они превращают Интернет в Интернет. Они позволяют нам связывать наши документы с любым другим документом (или ресурсом), с которым мы хотим. С их помощью мы также можем связывать документы с их конкретными частями, и мы можем сделать приложения доступными на простом веб-адресе (сравните это с локальными приложениями, которые должны быть установлены, и другими такими же вещами). Почти любой веб-контент может быть преобразован в ссылку, так что когда вы кликаете по ней (или иным образом активируете), она заставляет веб-браузер перейти на другой веб-адрес.

Анатомия ссылки

Простая ссылка создаётся путём обёртывания текста, который вы хотите превратить в ссылку, в элемент `<a>`, и придания этому элементу атрибута `href` (который также известен как гипертекстовая ссылка, или цель), который будет содержать веб-адрес, на который вы хотите указать ссылку.

```
<p>Я создал ссылку на  
  <a href="https://www.mozilla.org/ru/">домашнюю  
  страницу Mozilla</a>.  
</p>
```

Я создал ссылку на [домашнюю страницу Mozilla](https://www.mozilla.org/ru/).

Добавляем информацию через атрибут title

Другим атрибутом, который вы можете добавить к своим ссылкам, является — title. Он предназначен для хранения полезной информации о ссылке. Например, какую информацию содержит страница или другие вещи, о которых вам нужно знать. Например:

```
<p>Я создал ссылку на  
  <a href="https://www.mozilla.org/ru/"  
    title="Лучшее место для поиска дополнительной информации  
      о миссии Mozilla и о том, как внести свой вклад">домашнюю страницу  
Mozilla  
  </a> . Я создал ссылку на домашнюю страницу Mozilla .  
</p>
```

Лучшее место для поиска дополнительной информации
о миссии Mozilla и о том, как внести свой вклад

Ссылки-блоки

Как упоминалось ранее, вы можете превратить любой элемент в ссылку, даже блочный элемент. Если у вас есть изображение, которые вы хотели бы превратить в ссылку, вы можете просто поместить изображение между тегами `<a>`.

```
<a  
  href="https://github.com/">  
    
</a>
```

Домашняя работа

- Повторить всё что было в презентации