

Руководство по HTML5



HTML — «язык гипертекстовой разметки» — стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора. В HTML можно встроить программный код на языке программирования JavaScript, для управления поведением и содержанием веб-страниц. Также включение CSS в HTML позволяет задавать внешний вид и макет страницы. [Wikipedia](#)

Основы HTML

HTML — теговый язык разметки документов. Любой документ на языке HTML представляет собой набор элементов, причём начало и конец каждого элемента обозначается специальными пометками — тегами. Элементы могут быть пустыми, то есть не содержащими никакого текста и других данных. В этом случае обычно не указывается закрывающий тег (например, тег переноса строки `<br>` — одиночный и закрывать его не нужно). Кроме того, элементы могут иметь атрибуты, определяющие какие-либо их свойства (например, атрибут `href=«` у ссылки). Атрибуты указываются в открывающем теге. Вот примеры фрагментов HTML-документа:

- `<strong>`Текст между двумя тегами — открывающим и закрывающим.`</strong>`
- `<a href=«http://www.example.com»>`Здесь элемент содержит атрибут `href`, то есть гиперссылку.`</a>`
- А вот пример пустого элемента: `<br>`

Регистр, в котором набрано имя элемента и имена атрибутов, в HTML значения не имеет (в отличие от XHTML).

HTML-элементы

Элементы HTML являются строительными блоками HTML страниц. С помощью HTML разные конструкции, изображения и другие объекты, такие как интерактивная веб-форма, могут быть встроены в отображаемую страницу. HTML предоставляет средства для создания заголовков, абзацев, списков, ссылок, цитат и других элементов. Элементы HTML выделяются тегам, записанными с использованием угловых скобок. Такие теги, как `<img />` и `<input />`, напрямую вводят контент на страницу. Другие теги, такие как `<p>`, окружают и оформляют текст внутри себя и могут включать другие теги в качестве подэлементов. Браузеры не отображают HTML-теги, но используют их для интерпретации содержимого страницы.

[index.html](#)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
```

```
<title>Документ HTML5</title>
<link rel="stylesheet" type="text/css" href="style.css">
<script src="script.js"></script>
</head>
<body>
  <div>Содержание документа HTML5</div>
</body>
</html>
```

Структура HTML - документа	
Элемент	Описание
<!DOCTYPE html>	Объявление формата документа (это документ html с использованием HTML5)
<html> </html>	Является корневым элементом документа (между и находится все содержимое документа, выводимое в окне браузера)
<head> </head>	Техническая информация о документе (заголовок, описание, кодировку и т.д.). Введенная информация не отображается в браузере, но содержит данные, указывающие браузеру, как обрабатывать страницу.
<title> </title>	Наименование документа, которое отображается в строке заголовка веб-браузера
<meta>	Определяет метаинформацию страницы. В данном случае с помощью атрибута charset=«utf-8» указываем кодировку utf-8.
<style>	Задаёт стили, которые используются на странице. Для задания стилей в HTML-документе используется язык CSS.
<link>	Определяет взаимосвязь между текущей страницей и другими документами.
<script>	Позволяет присоединять к документу различные сценарии. Текст сценария может располагаться либо внутри этого элемента, либо во внешнем файле. Если текст сценария расположен во внешнем файле, то он подключается с помощью атрибутов элемента.
<body>	Предназначен для хранения содержания веб-страницы (контента), отображаемого в окне браузера. Информацию, которую следует выводить в документе, следует располагать именно внутри контейнера <body>
<div>	Блок, предназначенный для группировки контента.

Элемент <meta>

```
<meta name="description" content="Описание содержимого страницы">
<meta name="keywords" content="Ключевые слова через запятую">
```

```
<meta name="description" lang="ru" content="Описание содержимого страницы на русском">
<meta name="description" lang="en" content="Description">
<meta name="keywords" lang="ru" content="Ключевые слова через запятую">
<meta name="keywords" lang="en" content="Keywords">
```

• Поисковые машины

Индексация и переход по ссылкам разрешены:

```
<meta name="robots" content="index, follow">
```

Индексация разрешена, переход по ссылкам запрещен:

```
<meta name="robots" content="index, nofollow">
```

Индексация и переход по ссылкам запрещены:

```
<meta name="robots" content="noindex, nofollow">
```

я автоматической перезагрузки страницы через заданный промежуток времени (например, через 30 секунд) нужно воспользоваться значением refresh:

```
<meta http-equiv="refresh" content="30">
```

Чтобы перебросить посетителя на другую страницу, нужно указать URL-адрес в параметре url:

```
<meta http-equiv="refresh" content="0; url=http://yandex.ru/">
```

charset	Указывает кодировку символов для текущего HTML-документа: <code><meta charset=«UTF-8»></code>
content	Содержит произвольный текст, который определяет значение, ассоциируемое с атрибутом http-equiv или name, в зависимости от их значения.
http-equiv	Контролирует действия браузера на данной веб-странице (эквивалент HTTP заголовков). При отображении страницы браузер будет следовать инструкциям, заданным в атрибуте: default-style указывает предпочтительный стиль для использования на странице. Атрибут content должен содержать идентификатор элемента <code><link></code> , который ссылается на таблицу стилей CSS, или идентификатор элемента <code><style></code> , содержащего таблицу стилей. refresh указывает время в секундах до перезагрузки страницы или время до перенаправления на другую страницу, если в атрибуте content после указания времени идет строка <code>«url=адрес_страницы»</code> . Автоматическая перезагрузка страницы через заданный промежуток времени, в данном примере, через 30 секунд: <code><meta http-equiv=«refresh» content=«30»></code> Если необходимо сразу перебросить посетителя на другую страницу, то можно указать URL-адрес в параметре url: <code><meta http-equiv=«refresh» content=«0; url=http://mail.ru/»></code>
name	Ассоциируется со значением, содержащемся в атрибуте content. Не должен использоваться в случае, если для элемента уже заданы атрибуты http-equiv, charset или itemprop. application-name указывает название веб-приложения, используемого на странице. author указывает имя автора документа в свободном формате. description определяет краткое описание к содержимому страницы, например: <code><meta name=«description» content=«Описание содержимого страницы»></code> generator указывает один из пакетов программного обеспечения, используемого для создания документа, например: <code><meta name=«generator» content=«WordPress 4.0»></code> . keywords содержит список ключевых слов, разделенных запятыми, соответствующих содержимому страницы, например: <code><meta name=«keywords» content=«Ключевые слова через запятую»></code> . Также атрибут name может принимать следующие значения из расширенной спецификации, такие как creator, googlebot, publisher, robots, slurp, viewport, хотя ни одно из них еще не было официально принято.

HTML-атрибуты1.3

HTML-текст1.4

Элементы для заголовков Элементы для форматирования текста Элементы для ввода «компьютерного» текста Элементы для оформления цитат и определений Абзацы, средства переноса текста

HTML-ссылки1.5

Структура ссылки Абсолютный и относительный путь Якорь Как сделать изображение-ссылку Как сделать ссылку на телефонный номер, скайп или адрес электронной почты Атрибуты ссылок

HTML-изображения1.6

Элемент Адрес изображения Размеры изображения Форматы графических файлов Элемент <map> Элемент <area> Пример создания карты-изображения

HTML-таблицы1.7

Как создать таблицу Как создать строки (ряды) таблицы Как сделать ячейку заголовка столбца таблицы Как сделать ячейку тела таблицы Как добавить подпись (заголовок) к таблице Группирование строк и столбцов таблицы <colgroup> и <col> Группировка разделов таблицы <thead>, <tbody> и <tfoot> Как объединить ячейки таблицы Атрибуты элементов таблицы Пример создания таблицы

HTML-списки1.8

Маркированный список Нумерованный список Список определений <dl> Вложенный список Многоуровневый нумерованный список

Спецсимволы HTML1.9

Полезные знаки и символы Знаки пунктуации Стрелки Карточные масти Деньги Знаки зодиака

HTML-генераторы

```
<details>
<summary>Summary Text</summary>
Details Text
</details>
```

Summary Text

Details Text

Семантические элементы HTML5 1.11

Элемент документа 1.11.1

Метаданные документа 1.11.2

Элемент `<head>` Элемент `<title>` Элемент `<base>` Элемент `<link>` Элемент `<meta>` Элемент `<style>`

Разделы документа 1.11.3

Элемент `<body>` Элемент `<article>` Элемент `<section>` Элемент `<nav>` Элемент `<aside>`
Элементы `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` и `<h6>` Элемент `<header>` Элемент `<footer>`

Группировка содержимого 1.11.4

Элемент `<p>` Элемент `<address>` Элемент `<hr>` Элемент `<pre>` Элемент `<blockquote>`
Элемент `<ol>` Элемент `<ul>` Элемент `<li>` Элемент `<dl>` Элемент `<dt>` Элемент `<dd>` Элемент `<figure>`
Элемент `<figcaption>` Элемент `<main>` Элемент `<div>`

Семантика уровня текста 1.11.5

Элемент `<a>` Элемент `<em>` Элемент `<strong>` Элемент `<small>` Элемент `<s>` Элемент `<cite>`
Элемент `<q>` Элемент `<dfn>` Элемент `<abbr>` Элемент `<ruby>` Элемент `<rb>` Элемент `<rt>`
Элемент `<rtc>` Элемент `<rp>` Элемент `<data>` Элемент `<time>` Элемент `<code>` Элемент `<var>`
Элемент `<samp>` Элемент `<kbd>` Элементы `<sub>` и `<sup>` Элемент `<i>` Элемент `<b>`
Элемент `<u>` Элемент `<mark>` Элемент `<bdi>` Элемент `<bdo>` Элемент `<span>` Элемент `<br>`
Элемент `<wbr>`

Правки документа1.11.6

Элемент `<ins>` Элемент Атрибуты, общие для элементов `<ins>` и `<del>` ===== Встраиваемое содержимое1.11.7===== Элемент `<picture>` Элемент `<source>` Элемент `<img>` Элемент `<iframe>` Элемент `<embed>` Элемент `<object>` Элемент `<param>` Элемент `<video>` Элемент `<audio>` Элемент `<track>` Элемент `<map>` Элемент `<area>` ===== Ссылки1.11.8===== Ссылки, созданные элементами `<a>` и `<area>` Типы ссылок Примеры ссылок ===== Табличные данные1.11.9===== Элемент `<table>` Элемент `<caption>` Элемент `<colgroup>` Элемент `<col>` Элемент `<tbody>` Элемент `<thead>` Элемент `<tfoot>` Элемент `<tr>` Элемент `<td>` Элемент `<th>` ===== Формы1.11.10===== Элемент `<form>` Элемент `<label>` Элемент `<input>` Элемент `<button>` Элемент `<select>` Элемент `<datalist>` Элемент `<optgroup>` Элемент `<option>` Элемент `<textarea>` Элемент `<output>` Элемент `<progress>` Элемент `<meter>` Элемент `<fieldset>` Элемент `<legend>` Атрибут `autocomplete` ===== Интерактивные элементы1.11.11===== Элемент `<details>` Элемент `<summary>` Элемент `<dialog>` Скрипты1.11.12 Элемент `<script>` Элемент `<noscript>` Элемент `<template>` Элемент `<slot>` Элемент `<canvas>` ===== HTML5 аудио1.12===== Элемент `<audio>` Аудио кодеки Альтернативные медиа-ресурсы `<source>` Добавление субтитров и заголовков `<track>` Стилизованный пример аудио плеера ===== HTML5 видео1.13===== Элемент `<video>` Встраиваемый интерактивный контент `<embed>` Видеокодеки Видеоконтейнеры Альтернативные медиа-ресурсы `<source>` Добавление субтитров и заголовков `<track>` Пример: размещаем видео на сайте Видеоконвертеры Необязательные теги HTML5-разметки1.14 ===== HTML5 формы1.15===== Элемент `<form>` Группировка элементов формы `<fieldset>` Создание полей формы `<input>` Текстовые поля ввода `<textarea>` Раскрывающийся список `<select>` Надписи к полям формы `<label>` Кнопки `<button>` Флажки и переключатели в формах ===== Контентная модель HTML51.16===== Мета содержимое Потокное содержимое Секционное содержимое Заголовочное содержимое Текстовое содержимое Встроенное содержимое Интерактивное содержимое Явное содержимое Элементы, поддерживающие скрипт Корневое секционное содержимое Прозрачная модель содержимого



Read the [syntax](#) page¹⁾ again: «You can embed raw HTML code into your documents by using the HTML tags.» So to embed HTML you need to enclose the HTML with `<html></html>` (for inline HTML) or `<HTML></HTML>` (for block HTML). A little example: ~~The above syntax page has examples of what block vs inline look like.~~²⁾

```
<html><strong>This is my bold text</strong></html>
```

1) 2)

This section has been removed from the page and is only accessible through the page history.

From:
<https://wwoss.ru/> - worldwide open-source software

Permanent link:
https://wwoss.ru/doku.php?id=software:development:ps_pycharm:html5:html5&rev=1691749232

Last update: 2023/08/11 13:20

