

ДЛЯ ПРОФЕССИОНАЛОВ

HTML5

в примерах сообщества **Stack Overflow**

Издательство АСТ
г. Москва

УДК 004.43
ББК 32.973.2
Э34

*Последнюю версию книги «HTML5 Notes for Professionals»
на английском языке
можно скачать по адресу: <https://goalkicker.com/HTML5Book>.
Вы можете делиться этим PDF-документом со всеми,
кому он может быть полезен, бесплатно.
Книга создана на основе документации Stack Overflow
(<https://archive.org/details/documentation-dump.7z>), содержание которой
написано талантливыми участниками сообщества Stack Overflow.
Текстовые материалы распространяются на условиях лицензии
Creative Commons BY-SA.
Список авторов, внесших вклад в создание различных глав книги,
приведен в конце в разделе «Благодарности».
Изображения могут являться объектами авторского права
соответствующих владельцев, если не указано иное.*

HTML5 для профессионалов. Руководство по разработке в при-
Э34 мерах от сообщества Stack Overflow. — Москва : Издательство АСТ,
2025. — 160 с. : ил. — (Программирование. Оффер на миллион).
ISBN 978-5-17-178629-8.

Перед вами компактное, но вместе с тем весьма насыщенное руководство по HTML — основополагающему языку разметки, лежащему в основе каждой веб-страницы. Здесь собрана только самая важная и актуальная информация для быстрого и эффективного освоения HTML, от простейших структур до расширенных возможностей современной веб-разметки.

Вы узнаете, как создавать и структурировать контент с помощью заголовков, абзацев, списков, таблиц и мультимедийных элементов. Кроме того, рассматриваются принципы правильного использования ссылок, форм и управляющих элементов, организации навигации и семантической верстки. Особое внимание уделено атрибутам, работе с классами, идентификаторами, кастомными данными, а также взаимодействию HTML с CSS и JavaScript.

Книга охватывает современные возможности HTML5: адаптивные изображения, встроенные аудио- и видеоплееры, элемент <canvas> и SVG-графику, работу с ARIA-атрибутами для доступности, использование метаинформации и глобальных атрибутов. Приведены практические советы по оптимальной разметке, поддержке кросс-браузерности и улучшению UX.

Это издание станет надежной опорой как для начинающих программистов, делающих первые шаги в веб-разработке, так и для опытных специалистов, желающих систематизировать и расширить свои знания. Лаконичные объяснения, четкая структура и множество практических примеров делают эту книгу незаменимым инструментом в освоении и реальном применении HTML.

УДК 004.43
ББК 32.973.2

ISBN 978-5-17-178629-8

Перевод на русский язык: ООО «Интеджер».
Издание на русском языке: ООО «Издательство АСТ».

Содержание

Глава 1. Начало работы с HTML	9
1.1. Hello World.....	9
Глава 2. Доктайпы (Doctypes)	12
2.1. Добавление доктайпа	12
2.2. Доктайп HTML5	12
Глава 3. Заголовки	14
3.1. Использование заголовков.....	14
Глава 4. Абзацы	16
4.1. Абзацы в HTML	16
Глава 5. Форматирование текста	17
5.1. Выделение	17
5.2. Жирный, курсив и подчеркивание	18
5.3. Аббревиатура	19
5.4. Вставленный, удаленный или зачеркнутый текст	19
5.5. Верхний и нижний индекс.....	19
Глава 6. Якоря и гиперссылки	20
6.1. Ссылка на другой сайт.....	21
6.2. Ссылка на якорь	21
6.3. Ссылка на страницу того же сайта	22
6.4. Ссылка, которая набирает номер.....	23
6.5. Открытие ссылки в новой вкладке/окне.....	23
6.6. Ссылка, которая выполняет JavaScript.....	23
6.7. Ссылка, которая запускает почтовый клиент	24
Глава 7. Списки	26
7.1. Упорядоченный список.....	26
7.2. Неупорядоченный список.....	28
7.3. Вложенные списки	28
7.4. Список описаний (description list)	29
Глава 8. Таблицы	31
8.1. Простая таблица	31
8.2. Объединение столбцов или строк	31
8.3. Группы столбцов.....	33
8.4. Таблица с элементами thead, tbody,tfoot и caption	34
8.5. Область заголовка	35
Глава 9. Комментарии	38
9.1. Создание комментариев	38
9.2. Комментирование пробелов между строчными элементами.....	39

Глава 10. Классы и идентификаторы	40
10.1. Назначение элемента класса.....	40
10.2. Назначение элемента идентификатора.....	41
10.3. Допустимые значения.....	42
10.4. Проблемы, связанные с дублированием идентификаторов.....	45
Глава 11. Атрибуты данных (Data Attributes)	46
11.1. Поддержка в старых браузерах.....	46
11.2. Использование атрибутов данных (Data Attribute Use).....	46
Глава 12. Связывание ресурсов	47
12.1. JavaScript.....	47
12.2. Внешняя таблица стилей CSS.....	48
12.3. Favicon.....	49
12.4. Альтернативные таблицы стилей.....	49
12.5. Подсказки для ресурсов: dns-prefetch, prefetch, prerender.....	50
12.6. Атрибут media в теге <link>.....	51
12.7. Предыдущая и следующая страницы.....	51
12.8. Веб-фид (Web Feed).....	51
Глава 13. Включение JavaScript-кода в HTML	52
13.1. Обработка отключенного JavaScript.....	52
13.2. Ссылка на внешний файл JavaScript.....	53
13.3. Прямое включение JavaScript-кода.....	53
13.4. Включение JavaScript-файла с асинхронным выполнением.....	53
Глава 14. Использование HTML с CSS	54
14.1. Использование внешних таблиц стилей.....	54
14.2. Внутренние таблицы стилей.....	54
14.3. Встроенные стили.....	55
14.4. Несколько таблиц стилей.....	55
Глава 15. Изображения	57
15.1. Создание изображения.....	57
15.2. Выбор альтернативного текста.....	58
15.3. Адаптивное изображение с использованием атрибута srcset.....	59
15.4. Адаптивное изображение с использованием элемента <picture>.....	60
Глава 16. Карты изображений (Image Maps)	61
16.1. Введение в карты изображений.....	62
Глава 17. Элементы управления вводом	63
17.1. Текстовое поле.....	64
17.2. Флажки и переключатели.....	64
17.3. Валидация ввода.....	67
17.4. Цвет.....	69
17.5. Пароль.....	70
17.6. Файл.....	70
17.7. Кнопка.....	71
17.8. Отправка формы.....	73
17.9. Сброс формы.....	73

17.10. Скрытое поле (Hidden).....	73
17.11. Телефон.....	73
17.12. Электронная почта.....	74
17.13. Число.....	74
17.14. Диапазон.....	74
17.15. Поиск.....	75
17.16. Изображение.....	75
17.17. Неделя.....	75
17.18. URL.....	75
17.19. Локальные дата и время.....	75
17.20. Месяц.....	76
17.21. Время.....	76
17.22. Глобальные дата и время.....	76
17.23. Дата.....	77
Глава 18. Формы.....	78
18.1. Отправка данных.....	78
18.2. Атрибут target в tere form.....	79
18.3. Загрузка файлов.....	80
18.4. Группировка нескольких полей ввода.....	80
Глава 19. Элемент Div.....	82
19.1. Основное использование.....	82
19.2. Вложенность.....	83
Глава 20. Секционные элементы.....	84
20.1. Элемент <nav>.....	84
20.2. Элемент <article>.....	85
20.3. Элемент <main>.....	87
20.4. Элемент <header>.....	89
20.5. Элемент <footer>.....	89
20.6. Элемент <section>.....	90
Глава 21. Навигационные панели.....	91
21.1. Базовая навигационная панель.....	91
21.2. Навигационная панель в HTML5.....	91
Глава 22. Элемент <label>.....	92
22.1. Об элементе <label>.....	92
22.2. Базовое использование.....	93
Глава 23. Элемент Output (Вывод).....	94
23.1. Использование атрибутов For и Form в элементе Output.....	94
23.2. Элемент Output с атрибутами.....	95
Глава 24. Пустые элементы (Void Elements).....	96
24.1. Пустые элементы.....	96
Глава 25. Медиаэлементы.....	98
25.1. Аудио.....	98
25.2. Видео.....	98
25.3. Использование элементов <video> и <audio> для отображения аудио/видео контента....	99
25.4. Видео в качестве заголовка или фона.....	100

Глава 26. Элемент Progress	101
26.1. Прогресс	101
26.2. Изменение цвета полосы прогресса	101
26.3. Запасной вариант с использованием HTML.....	102
Глава 27. Элементы управления выпадающими меню	103
27.1. Элемент <select>	103
27.2. Опции.....	104
27.3. Группы опций.....	105
27.4. Элемент <datalist>	105
Глава 28. Элемент <embed>	107
28.1. Основное использование	107
28.2. Определение MIME-типа	107
Глава 29. IFrames (Встроенные фреймы)	108
29.1. Основы встроенного фрейма	108
29.2. Песочница (Sandboxing).....	108
29.3. Установка размера фрейма.....	109
29.4. Использование атрибута srcdoc	109
29.5. Использование якорей с IFrames	110
Глава 30. Языки контента	111
30.1. Основной язык документа.....	111
30.2. Язык элемента	111
30.3. Элементы с несколькими языками	111
30.4. Региональные URL	112
30.5. Обработка атрибутов с разными языками	112
Глава 31. SVG	113
31.1. Встроенный SVG.....	113
31.2. Встраивание внешних SVG-файлов в HTML	114
31.3. Встраивание SVG с помощью CSS.....	114
Глава 32. Canvas (Холст)	116
32.1. Базовый пример.....	116
32.2. Рисование двух прямоугольников на <canvas>	117
Глава 33. Метаинформация	119
33.1. Информация о странице	119
33.2. Кодировка символов	120
33.3. Роботы (robots).....	120
33.4. Социальные сети	121
33.5. Управление мобильной версткой	123
33.6. Автоматическое обновление	124
33.7. Распознавание телефонных номеров	125
33.8. Автоматическое перенаправление	125
33.9. Веб-приложение	125
Глава 34. Разметка компьютерного кода	126
34.1. Блок с использованием <pre> и <code>	126
34.2. Встроенный код с использованием <code>	126

Глава 35. Разметка цитат	127
35.1. Встроенные цитаты с использованием <q>	127
35.2. Блочные цитаты с использованием <blockquote>	127
Глава 36. Tabindex	130
36.1. Добавление элемента в порядок табуляции.....	130
36.2. Удаление элемента из порядка табуляции	130
36.3. Определение пользовательского порядка табуляции (не рекомендуется)	130
Глава 37. Глобальные атрибуты	132
37.1. Атрибут contenteditable	132
Глава 38. Кеш HTML5	134
38.1. Базовый пример кеша HTML5	134
Глава 39. Атрибуты событий HTML	135
39.1. События форм HTML	135
39.2. События клавиатуры	135
Глава 40. Символьные сущности (Character Entities)	136
40.1. Символьные сущности в HTML.....	136
40.2. Распространенные специальные символы	136
Глава 41. ARIA	138
41.1. role="presentation"	138
41.2. role="alert"	138
41.3. role="alertdialog".....	138
41.4. role="application"	139
41.5. role="article"	139
41.6. role="banner".....	140
41.7. role="button".....	140
41.8. role="cell".....	140
41.9. role="checkbox"	140
41.10. role="columnheader".....	141
41.11. role="combobox"	141
41.12. role="complementary"	141
41.13. role="contentinfo"	141
41.14. role="definition".....	142
41.15. role="dialog".....	142
41.16. role="directory".....	142
41.17. role="document"	142
41.18. role="form"	143
41.19. role="grid"	144
41.20. role="gridcell".....	144
41.21. role="group".....	144
41.22. role="heading"	145
41.23. role="img".....	145
41.24. role="link".....	145
41.25. role="list"	145
41.26. role="listbox"	145

41.27. role="listitem"	146
41.28. role="log"	146
41.29. role="main"	146
41.30. role="marquee"	146
41.31. role="math"	147
41.32. role="menu"	147
41.33. role="menubar"	147
41.34. role="menuitem"	147
41.35. role="menuitemcheckbox"	148
41.36. role="menuitemradio"	148
41.37. role="navigation"	148
41.38. role="note"	148
41.39. role="option"	149
41.40. role="progressbar"	149
41.41. role="radio"	149
41.42. role="region"	149
41.43. role="radiogroup"	149
41.44. role="row"	150
41.45. role="rowgroup"	150
41.46. role="rowheader"	150
41.47. role="scrollbar"	151
41.48. role="search"	151
41.49. role="searchbox"	151
41.50. role="separator"	152
41.51. role="slider"	152
41.52. role="spinbutton"	152
41.53. role="status"	152
41.54. role="switch"	152
41.55. role="tab"	153
41.56. role="table"	153
41.57. role="tablist"	153
41.58. role="tabpanel"	153
41.59. role="textbox"	154
41.60. role="timer"	154
41.61. role="toolbar"	154
41.62. role="tooltip"	154
41.63. role="tree"	155
41.64. role="treegrid"	155
41.65. role="treeitem"	155
Благодарности	157

Глава 1.

Начало работы с HTML

Версия	Дата выпуска
1.0	01-01-1994
2.0	24-11-1995
3.2	14-01-1997
4.0	24-04-1998
4.01	24-12-1999
5	29-10-2014
5.1	01-11-2016
5.2	14-12-2017
5.3	24-01-2018

1.1. Hello World

Введение

HTML (Hypertext Markup Language, язык гипертекстовой разметки) использует систему разметки, состоящую из элементов, которые представляют определенное содержимое. *Разметка* означает следующее: с помощью HTML вы объявляете, что именно будет показано зрителю, а не как это будет показано. Визуальное представление определяется с помощью каскадных таблиц стилей (CSS) и реализуется браузерами. Некоторые элементы, такие как ``, которые все еще существуют, полностью устарели и не должны использоваться авторами^[1].

HTML иногда называют языком программирования, но он не содержит логики, поэтому является языком разметки. HTML-теги обеспечивают семантическое значение и машинную читаемость содержимого на странице.

Элемент обычно состоит из открывающего тега (`<element_name>`), закрывающего тега (`</element_name>`), которые содержат имя элемента, окруженное угловыми скобками, и содержимого между ними:

```
<element_name>...content...</element_name>
```

Существуют некоторые HTML-элементы, которые не имеют закрывающего тега или содержимого. Они называются пустыми элементами (void elements). К ним относятся ``, `<meta>`, `<link>` и `<input>`.

Имена элементов можно рассматривать как описательные ключевые слова для содержимого, которое они содержат, например, `video` (видео), `audio` (аудио), `table` (таблица), `footer` (футер).

HTML-страница может состоять из сотен элементов, которые затем читаются веб-браузером, интерпретируются и преобразуются в читаемое или озвученное содержимое на экране.

Для этого документа важно отметить разницу между элементами и тегами:

- Элементы: `video`, `audio`, `table`, `footer`
- Теги: `<video>`, `<audio>`, `<table>`, `<footer>`, `</html>`, `</body>`

Понимание элементов

Давайте разберем, что представляет собой тег.

Тег `<p>` представляет собой обычный абзац (paragraph).

Элементы чаще всего имеют открывающий и закрывающий теги. Открывающий тег содержит имя элемента в угловых скобках (`<p>`). Закрывающий тег идентичен открывающему, но с добавлением косой черты (/) между открывающей угловой скобкой и именем элемента (`</p>`).

Содержимое может быть помещено между этими двумя тегами:

```
<p>This is a simple paragraph.</p>.
```

Создание простой страницы

Следующий пример HTML создает простую веб-страницу «Hello World».

HTML-файлы можно создавать с помощью любого текстового редактора. Файлы должны быть сохранены с расширением `.html` или `.htm[2]`, чтобы они распознавались как HTML-файлы.

После создания этот файл можно открыть в любом веб-браузере.

```
<!DOCTYPE html>
<html lang="en"> <!-- Язык документа – английский -->
<head>
  <meta charset="UTF-8"> <!-- Кодировка символов – UTF-8 -->
  <title>Hello!</title> <!-- Заголовок страницы -->
</head>
<body>
  <h1>Hello World!</h1> <!-- Заголовок первого уровня -->
  <p>This is a simple paragraph.</p> <!-- Простой абзац -->
</body>
</html>
```

Разбор простой страницы

Ниже поясняются теги, использованные в примере.

Тег	Значение
<!DOCTYPE>	Определяет версию HTML, используемую в документе. В данном случае это HTML5.
<html>	Открывает страницу. Никакая разметка не должна идти после закрывающего тега (</html>). Атрибут lang объявляет основной язык страницы с использованием кодов ISO (например, en для английского).
<head>	Открывает раздел head, который не отображается в основном окне браузера, но содержит информацию о HTML-документе, называемую метаданными. Он также может содержать импорт внешних таблиц стилей и скриптов. Закрывающий тег — </head>.
<meta>	Предоставляет браузеру метаданные о документе. Атрибут charset объявляет кодировку символов. Современные HTML-документы всегда должны использовать UTF-8, даже если это не является обязательным требованием. В HTML тег <meta> не требует закрывающего тега.
<title>	Заголовок страницы. Текст, написанный между этим открывающим и закрывающим тегом (</title>), будет отображаться на вкладке страницы или в заголовке браузера.
<body>	Открывает часть документа, отображаемую пользователям, то есть все видимое или озвученное содержимое страницы. Никакое содержимое не должно добавляться после закрывающего тега </body>.
<h1>	Заголовок первого уровня для страницы.
<p>	Представляет собой обычный абзац текста.

1. HTML5, 11.2 Несоответствующие функции (<https://html.spec.whatwg.org/multipage/obsolete.html#non-conforming-features>).



2. .htm унаследован от ограничения DOS на трехсимвольные расширения файлов. (<https://en.wikipedia.org/wiki/DOS>).



Глава 2.

Доктайпы (Doctypes)

Доктайпы (Doctypes) — сокращение от document type (тип документа) — помогают браузерам понять версию HTML, на которой написан документ, для лучшей интерпретации. Объявления доктайпов не являются HTML-тегами и должны располагаться в самом верху документа. В этой теме объясняются структура и объявление различных доктайпов в HTML.

2.1. Добавление доктайпа

Объявление `<!DOCTYPE>` всегда должно быть включено в начало HTML-документа, перед тегом `<html>`.

Версия ≥ 5

См. раздел «Доктайп HTML5» ниже, чтобы подробнее узнать о доктайпе HTML5.

2.2. Доктайп HTML5

HTML5 не основан на SGML (Standard Generalized Markup Language, стандартный обобщенный язык разметки) и поэтому не требует ссылки на DTD (Document Type Definition, определение типа документа).

Объявление доктайпа HTML5:

```
<!DOCTYPE html>
```

Регистронезависимость

Согласно спецификации W3.org HTML5 DOCTYPE Доктайп должен состоять из следующих компонентов в указанном порядке:

1. Строка, которая является регистронезависимым совпадением строки «`<!DOCTYPE`».

Следовательно, следующие доктайпы также являются допустимыми:

```
<!doctype html>  
<!d0CtYpe html>  
<!DocTYpe html>
```

Спецификации W3.org HTML5 DOCTYPE (<https://html.spec.whatwg.org/multipage/syntax.html#the-doctype>).



В этой статье на Stack Overflow подробно обсуждается тема Uppercase or lowercase doctype? (<https://stackoverflow.com/questions/7020961/uppercasse-or-lowercasse-doctype>)



Глава 3.

Заголовки

HTML предоставляет не только обычные теги для абзацев, но и шесть отдельных тегов заголовков для обозначения заголовков различных размеров и толщины. Они нумеруются от заголовка 1 до заголовка 6, где заголовок 1 имеет самый крупный и жирный текст, а заголовок 6 — самый мелкий и тонкий, вплоть до уровня абзаца. В данной главе подробно описывается правильное использование этих тегов.

3.1. Использование заголовков

Заголовки могут использоваться для описания темы, которую они предвещают, и определяются с помощью тегов `<h1>`–`<h6>`. Заголовки поддерживают все глобальные атрибуты.

- `<h1>` определяет самый важный заголовок.
- `<h6>` определяет наименее важный заголовок.

Пример определения заголовков:

```
<h1>Heading 1</h1> <!-- Заголовок 1 -->
<h2>Heading 2</h2> <!-- Заголовок 2 -->
<h3>Heading 3</h3> <!-- Заголовок 3 -->
<h4>Heading 4</h4> <!-- Заголовок 4 -->
<h5>Heading 5</h5> <!-- Заголовок 5 -->
<h6>Heading 6</h6> <!-- Заголовок 6 -->
```

Правильная структура важна

Поисковые системы и другие пользовательские агенты обычно индексируют содержимое страницы на основе элементов заголовков, например, для создания оглавления, поэтому использование правильной структуры для заголовков имеет большое значение.

Как правило, статья должна содержать один элемент `<h1>` для основного заголовка, за которым следуют подзаголовки `<h2>` — при необходимости можно углубляться на уровень ниже. Если элементы `<h1>` используются на более высоком уровне, они не должны описывать содержимое более низкого уровня.

Пример документа (дополнительные отступы для иллюстрации иерархии):

```
<h1>Main title</h1> <!-- Основной заголовок -->
<p>Introduction</p> <!-- Введение -->

  <h2>Reasons</h2> <!-- Причины -->

    <h3>Reason 1</h3> <!-- Причина 1 -->
    <p>Paragraph</p> <!-- Абзац -->

    <h3>Reason 2</h3> <!-- Причина 2 -->
    <p>Paragraph</p> <!-- Абзац -->

  <h2>In conclusion</h2> <!-- В заключение -->
  <p>Paragraph</p> <!-- Абзац -->
```