

Элемент

На этой странице собраны все элементы HTML. Они сгруппированы по функциям, чтобы помочь вам найти то, что вам нужно для реализации ваших идей. Хотя это руководство написано для относительных новичков в программировании, мы хотим, чтобы оно было полезным для всех.

Основные элементы

Основные элементы являются основой любого HTML документа. Вы увидите эти элементы в исходном коде для всех веб-страниц после задания типа документа на первой строке на странице. DOCTYPE определяет, какую версию (X) HTML эта страница использует. Элементы страницы находятся между открывающим тегом <HTML> и закрывающим </ HTML>. Элемент <html> называется корневым элементом.

Элемент	Описание
<html>	Представляет собой корень (элемент верхнего уровня) HTML-документа, поэтому его так же называют корневым элементом. Все другие элементы должны быть потомками этого элемента (должны находиться внутри него).

Метаданные содержат информацию о странице. Они включают в себя информацию о стилях, скрипты и данные, чтобы помочь программному обеспечению (поисковые системы, браузеры и т.д.) использовать и отображать страницу. Метаданные для стилей и скрипты могут быть определены на странице или ссылке на другой файл, который имеет информацию.

Элемент	Описание
<base>	Определяет основной адрес (URL), используемый для всех относительных адресов (URLs) в документе. Может быть только один <base> элемент в одном документе.
<head>	Содержит машиночитаемую информацию (metadata) о документе, например его заголовок, скрипты и страницы стилей.
<link>	Определяет отношения между текущим документом и внешним ресурсом. Этот элемент чаще всего используется для ссылки на stylesheets, а также для создания иконок сайта (как для иконок в стиле «favicon», так и для иконок домашних экранов и приложений мобильных устройств) среди прочего.
<meta>	Представляет такие метаданные, которые не могут быть представлены другими HTML-метатегами, такими как <base>, <link>, <script>, <style> или <title>.
<style>	Содержит стилевую информацию для документа или его части. По умолчанию стилевые инструкции внутри этого элемента считаются написанными на CSS.
<title>	Определяет заголовок документа, который отображается в заголовке окна браузера или на вкладке страницы. Он содержит только текст, а теги внутри элемента игнорируются.

Секционирование корня

Элемент Описание <body> Представляет собой контент (содержимое) документа HTML. В документе может быть только один элемент <body>. Секционирование содержания Секционирование содержания элементов позволяет разложить содержимое документа на логические части. Используйте секционирование элементов для создания общих черт

содержания страницы, включая шапку и подвал и заголовочные элементы для обозначения разделов.

Элемент Описание `<address>` Задаёт контактные данные для ближайшего родительского `<article>` или `<body>` ; В последнем случае применяется ко всему документу, визуально выделяется курсивом. `<article>` Представляет самостоятельную часть документа, страницы, приложения или сайта, предназначенную для независимого распространения или повторного использования. Этот элемент может представлять статью на форуме, статью в журнале или газете, запись в блоге или какой-либо другой самостоятельный фрагмент содержимого. `<aside>` Представляет собой часть документа, чьё содержимое только косвенно связано с основным содержимым документа. Чаще всего представлен в виде боковой панели, сносок или меток. `<footer>` Представляет собой нижний колонтитул (футер, подвал) для своего ближайшего секционного контента или секционного корня. Футер обычно содержит информацию об авторе раздела, информацию об авторском праве или ссылки на связанные документы. `<header>` Представляет собой вводный контент, обычно группу вводных или навигационных средств. Он может содержать другие элементы-заголовки, а также логотип, форму поиска, имя автора и другие элементы. `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>` HTML-элементы `<h1>`-`<h6>` представляют собой 6 уровней заголовков секций. `<h1>` это наибольший заголовок и `<h6>` - наименьший `<main>` Предназначен для основного контента (содержимого) `<body>` документа (страницы). Основной контент состоит из контента, который непосредственно относится к главной теме документа или её развивает. `<nav>` Определяет отдельную секцию документа, назначение которой обозначение ссылок навигации (как внутри текущего документа, так и ведущих на другую страницу). В качестве примера такой секции можно привести меню, якорные ссылки. `<section>` Представляет собой автономный раздел — который не может быть представлен более точным по семантике элементом — внутри HTML-документа. Как правило, но не всегда, разделы имеют заголовок.

Текстовое содержание

Используйте текстовые элементы HTML для организации блоков или разделов содержания, размещённые между тегам открытия `<body>` и закрытия `</body>`. Важное значение для accessibility и SEO, чтобы эти элементы определили цель или структуру этого контента.

Элемент Описание `<blockquote>` Указывает на то, что заключённый в нем текст является развёрнутой цитатой. Обычно он (текст) визуально выделяется наклонным (смотри Примечание (en-US), где говорится о том, как это изменить). URI на источник цитаты можно указать в атрибуте `cite`, тогда как текстовое представление источника может быть задано элементом `<cite>`. `<dd>` Предоставляет подробности или определение предшествующего термина (`<dt>`) в списке определений (`<dl>`). `<div>` Является универсальным контейнером для потокового контента. Он не влияет на контент или макет до тех пор, пока не будет стилизован с помощью CSS. `<dl>` Представляет собой список описаний. Этот элемент служит контейнером для списка пар терминов (определяемых элементом `<dt>`) и их описаний (определяемых элементами `<dd>`). Этот элемент обычно используют при создании глоссария или для отображения метаданных (списка пар ключ-значение). `<dt>` Определяет термин в описании или списке определений, как таковой должен использоваться внутри элемента `<dl>` . Обычно за ним следует элемент `<dd>`. Кроме того, несколько элементов `<dt>` идущие друг за другом будут содержать своё определение в следующем идущем за ним элементом `<dd>`. `<figcaption>` Представляет собой подпись (заголовок) или легенду, описывающую остальную часть содержимого родительского элемента `<figure>`. `<figure>` Представляет самостоятельный

контент, часто с подписью (заголовком), которая указывается с помощью элемента (`<figcaption>`). Диаграмма и её подпись представляет собой единое целое. `<hr>` Служит для тематического разделения абзацев. Рисует горизонтальную прямую `` Используется для создания элементов списка. Он также должен находиться в родительском элементе: упорядоченном списке (``), неупорядоченном списке (``), или меню (`<menu>`). В меню и неупорядоченных списках, элементы списка обычно отображаются с маркерами в виде круга. В упорядоченных списках, они обычно отображаются с маркерами в виде возрастающего счётчика слева, например числа или буквы. `<menu>` Представляет группу команд, которые пользователь может выполнить или активировать. Он включает как меню-списки, которые могут отображаться в верхней части экрана, так и контекстные меню, например, такие, что могут появиться под кнопкой после нажатия. `` Используется для упорядоченного списка — в частности для пронумерованного списка. `<p>` Представляет собой абзац. Абзацы обычно представлены в визуальной среде в виде блоков текста, отделённых от соседних блоков пустыми строками и/или отступом в первой строке, кроме этого HTML-абзацы могут представлять собой структурную группировку однотипного содержимого, например изображений или полей формы. `<pre>` Представляет собой предварительно отформатированный текст, который должен быть представлен точно так, как написано в HTML-файле. Текст обычно отображается с использованием непропорционального («monospace») шрифта. Пробелы внутри этого элемента отображаются как записанные. `` Используется для неупорядоченного списка - в частности для маркированного списка.

Встроенные текстовые семантики

Используйте встроенную в HTML текстовую семантику, чтобы определить смысл, структуру или стиль текста, линий или части текста.

Элемент Описание `<a>` Определяет гиперссылку для перехода на определённое место на странице или на другую страницу в Интернете. Также он может быть использован (в устаревшем варианте) для создания якоря — это место назначения для гиперссылок внутри страницы: так ссылки не ограничены только в перемещении между страницами. `<abbr>` Представляет собой аббревиатуру и, опционально, может показывать её расшифровку. Расшифровка записывается в атрибут `title`. Иные значения `title`, кроме расшифровки аббревиатуры не допускаются. `` Является частью текста стилистически отличающейся от нормального текста, не носящий какого-либо специального значения или важности, и как правило выделен жирным шрифтом. `<bdi>` Изолирует от окружающего текста текстовый фрагмент, направление в котором может отличаться от направления окружающего текста (но не обязательно отличается). `<bdo>` Переопределяет текущее направление текста так, что текст внутри отображается в другом направлении. `
` Устанавливает перевод строки в тексте (возврат каретки). Он полезен при написании поэмы или адреса, где важно деление на строки. `<cite>` Представляет из себя ссылку на источник цитаты. Он должен включать в себя название произведения или URL, который может быть в сокращённом виде в соответствии с правилами, используемых для добавления метаданных цитирования. `<code>` Отображает его содержимое в стиле, предназначенном для указания на то, что текст является коротким фрагментом компьютерного кода. По умолчанию текст содержимого отображается с использованием шрифта монопространства по умолчанию пользовательского агента. `<data>` *Связывает данное содержимое с машиночитаемым представлением. Если содержимое относится к времени или дате, необходимо использовать элемент `<time>`. `<dfn>` Используется для указания термина, определяемого в контексте фразы или предложения. Элемент `<p>`, пара `<dt>/<dd>` или элемент `<section>`, который является ближайшим предком `<dfn>` считается определением термина. `` Отмечает акцентированный текст. Элемент ``

может быть вложенным, причём каждый уровень вложенности указывает на большую степень акцента. `<i>` (en-US) Представляет собой отрывок текста, выделяемый среди прочего по определенным причинам, например: идиоматический текст, технические термины, систематические обозначения и прочее. Обычно отмечается курсивом, от чего и получил свое название (от англ. *Italic*). `<kbd>` Указывает на то, что текст внутри элемента описывает пользовательский ввод с клавиатуры, голосовой ввод или ввод с использованием любого другого типа устройств ввода текста. По общему соглашению, user agent использует моноширинный шрифт для отображения содержимого элемента `<kbd>` по умолчанию, хотя это и не указывается HTML стандартом явно. `<mark>` Представляет текст, выделенный в справочных целях из-за своей актуальности в определённом контексте. Например, он может быть использован на странице с результатом поиска, в которой выделяется каждый экземпляр искомого слова. `<q>` (en-US) Indicates that the enclosed text is a short inline quotation. Most modern browsers implement this by surrounding the text in quotation marks. This element is intended for short quotations that don't require paragraph breaks; for long quotations use the `<blockquote>` element. `<rp>` (en-US) Used to provide fall-back parentheses for browsers that do not support display of ruby annotations using the `<ruby>` element. One `<rp>` element should enclose each of the opening and closing parentheses that wrap the `<rt>` (en-US) element that contains the annotation's text. `<rt>` (en-US) Specifies the ruby text component of a ruby annotation, which is used to provide pronunciation, translation, or transliteration information for East Asian typography. The `<rt>` element must always be contained within a `<ruby>` element. `<ruby>` Предназначен для добавления небольшой аннотации сверху или снизу от заданного текста. Такая форма записи преимущественно используется для идеографической письменности вроде китайского языка, но может применяться и для других языков, если требуется написать один текст над другим. `<s>` (en-US) Renders text with a strikethrough, or a line through it. Use the `<s>` element to represent things that are no longer relevant or no longer accurate. However, `<s>` is not appropriate when indicating document edits; for that, use the `del` and `ins` elements, as appropriate. `<samp>` (en-US) Used to enclose inline text which represents sample (or quoted) output from a computer program. Its contents are typically rendered using the browser's default monospaced font (such as Courier or Lucida Console). `<small>` (en-US) Represents side-comments and small print, like copyright and legal text, independent of its styled presentation. By default, it renders text within it one font-size smaller, such as from small to x-small. `` Является основным строковым контейнером для фразового контента, который, по существу, ничего не представляет. Он может использоваться для группировки элементов в целях стилизации (использование атрибутов `class` или `id`) или потому, что они имеют общие значения атрибутов, например `lang`. Он должен быть использован только когда нет другого подходящего по семантике элемента. `` очень похож на элемент `<div>`, но `<div>` является блочным элементом, в то время как `` является строчным. `` Указывает на то, что его содержание имеет большое значение, серьёзность или срочность. Браузеры обычно отображают содержимое жирным шрифтом. `<sub>` (en-US) Specifies inline text which should be displayed as subscript for solely typographical reasons. Subscripts are typically rendered with a lowered baseline using smaller text. `<sup>` (en-US) Specifies inline text which is to be displayed as superscript for solely typographical reasons. Superscripts are usually rendered with a raised baseline using smaller text. `<time>` Используется для представления либо времени в 24-часовом формате, либо точной даты по Григорианскому календарю (с опциональным указанием времени и часового пояса). `<u>` The HTML Unarticulated Annotation Element (`<u>`) represents a span of inline text which should be rendered in a way that indicates that it has a non-textual annotation. This is rendered by default as a simple solid underline, but may be altered using CSS. `<var>` (en-US) Represents the name of a variable in a mathematical expression or a programming context. It's typically presented using an italicized version of the current typeface, although that behavior is browser-dependent. `<wbr>` Предоставляет возможность переноса слова - позицию в тексте, где браузер может по желанию разбить строку, в

противном случае его правила разрыва строки не будут создавать разрыв в этом месте. Изображения и мультимедиа HTML позволяет использовать различные мультимедийные ресурсы, такие как изображения, аудио и видео.

Элемент Описание <area> Определяет активную область на изображении и, при желании, связывает её с гипертекстовой ссылкой (en-US). Этот элемент используется только внутри элемента <map>. <audio> Используется для встраивания звукового контента в документ. Он может содержать один или более источников аудио, представленных с помощью атрибута src или элемента <source> - браузер выберет один наиболее подходящий. Он так же может предназначаться для потокового мультимедиа, используя интерфейс MediaStream. Встраивает изображение в документ. Это замещаемый элемент. <map> Используется с элементами <area> для определения карты изображения (интерактивной области ссылок). <track> Используется как дочерний элемент медиа-элементов <audio> and <video>. Позволяет указать синхронизированные текстовые дорожки (или данные на основе времени), например, для автоматической обработки субтитров. Файлы треков используют формат WebVTT (.vtt файлы) — Web Video Text Tracks или Timed Text Markup Language (TTML). <video> Для встраивания видео контента в документ используйте элемент HTML <video>. Видео элемент может содержать один или несколько источников видео. Чтобы указать источник видео, необходимо использовать атрибут src или элемент <source>; браузер сам определит наиболее подходящий источник. Встроенное содержание Содержание мультимедийных HTML включает в себя другие материалы, даже если они не всегда легко взаимодействуют.

Элемент Описание <embed> Вставляет расширенный контент в выбранное место документа. Этот контент может быть представлен от внешнего приложения или другого источника интерактивного контента, такого как плагин для браузера, например. <iframe> (en-US) Represents a nested browsing context, embedding another HTML page into the current one. <object> (en-US) Represents an external resource, which can be treated as an image, a nested browsing context, or a resource to be handled by a plugin. <picture> Служит контейнером для одного или более элементов <source> и одного элемента для обеспечения оптимальной версии изображения для различных размеров экрана. Браузер рассмотрит каждый из дочерних элементов <source> и выберет один, соответствующий лучшему совпадению; если совпадений среди элементов <source> найдено не будет, то будет выбран файл, указанный атрибутом src элемента . Затем выбранное изображение отображается в пространстве, занятом элементом . <portal> (en-US) Enables the embedding of another HTML page into the current one for the purposes of allowing smoother navigation into new pages. <source> Указывает несколько медиа-ресурсов для элементов <picture>, <video> и <audio>. Это пустой элемент. Он обычно используется для обслуживания одного и того же медиа-контента в нескольких форматах, поддерживаемых различными браузерами (en-US). SVG and MathML You can embed SVG and MathML content directly into HTML documents, using the <svg> and <math> elements.

Элемент Описание <svg> Представляет собой контейнер, который определяет новую систему координат и область просмотра. Он используется, как самый внешний элемент документов SVG, но также может использоваться для встраивания фрагмента SVG в документ SVG или HTML. <math> Элементом верхнего уровня в MathML является тэг <math>. Каждый допустимый экземпляр MathML должен быть внутри этого контейнера. Он не допускает вложений, но внутри может быть произвольное число других дочерних элементов. Скриптинг Чтобы создавать динамический контент и веб-приложения, HTML поддерживает использование скриптовых языков, наиболее известным является JavaScript. Некоторые элементы поддерживают эту возможность.

Элемент Описание <canvas> Может быть использован для отрисовки графики при помощи скриптов (обычно используется JavaScript). Например, его можно использовать для отрисовки

графиков, делать композиции фото или даже выполнять анимации. Вы можете (и должны) дать альтернативное содержание внутри блока `<canvas>`. Этот контент будет рендериться в обоих браузерах, в старых которые не поддерживают `canvas` и в браузерах с отключённым JavaScript. `<noscript>` Определяет секцию HTML кода, которая будет вставлена, если в браузере пользователя нет либо отключена поддержка JavaScript. `<script>` Используется для встраивания или подключения исполняемого JavaScript кода. Элемент `<script>` также может использоваться с другими языками, такими как GLSL (en-US) от WebGL. Разграничительные правки Эти элементы позволяют вам отметить определённые части текста.

Элемент Описание `` Представляет диапазон текста, который был удалён из документа. Он может быть использован, например, при отображении «отслеживания изменений» или различий в исходном коде. **Элемент `<ins>`** можно использовать для противоположной цели: указание текста, который был добавлен в документ. `<ins>` Представляет собой диапазон текста, который был добавлен в документ **Содержание таблиц** Эти элементы используются для создания и обработки табличных данных.

Элемент Описание `<caption>` Определяет название (заголовок) таблицы. Если этот элемент используется, он всегда должен быть первым вложенным элементом тэга `<table>` (en-US). Внешний вид и расположение заголовка по отношению к самой таблице может быть изменено с помощью стилей CSS `caption-side` (en-US) and `text-align`. **`<col>`** Определяет столбец в таблице и используется для определения общей семантики на всех ячейках. Обычно он находится в элементе `<colgroup>` (en-US). `<colgroup>` (en-US) Defines a group of columns within a table. `<table>` (en-US) Represents tabular data — that is, information presented in a two-dimensional table comprised of rows and columns of cells containing data. `<tbody>` (en-US) Encapsulates a set of table rows (`<tr>` (en-US) elements), indicating that they comprise the body of the table (`<table>` (en-US)). `<td>` Определяет ячейку таблицы которая содержит данные. Участвует в табличной модели. `<tfoot>` Определяет набор строк суммирующих столбцы таблицы. `<th>` (en-US) Defines a cell as header of a group of table cells. The exact nature of this group is defined by the scope and headers attributes. `<thead>` (en-US) Defines a set of rows defining the head of the columns of the table. `<tr>` (en-US) Defines a row of cells in a table. The row's cells can then be established using a mix of `<td>` (data cell) and `<th>` (en-US) (header cell) elements. **Формы HTML** содержит некоторое количество элементов, которые используются вместе для создания форм, которые пользователь может заполнить и отправить на сервер. Доступно множество информации по этой теме в руководстве по HTML формам.

Элемент Описание `<button>` Создаёт интерактивную кнопку, которая может быть использована в формах или в любом другом месте документа, который требует простой, стандартной кнопки. По умолчанию, кнопки HTML обычно представлены в стиле, аналогичном стилю хост-платформы, на которой работает user agent, но вы можете изменить внешний вид кнопки, используя CSS. **`<datalist>`** Содержит набор опций (`<option>`), доступных для выбора. Выбранное значение будет установлено для элемента `<input>`, с атрибутом `list`. **`<fieldset>`** Используется для группировки нескольких элементов управления в веб-форме. **`<form>`** Представляет (собой) раздел документа, содержащий интерактивные элементы управления, которые позволяют пользователю отправлять информацию на веб-сервер. **`<input>`** Используется для создания интерактивных элементов управления в веб-формах для получения данных от пользователя; в зависимости от устройства и user agent, доступен широкий выбор типов входных данных и виджетов управления. Из-за огромного количества возможных сочетаний типов ввода и атрибутов это один из самых мощных и сложных элементов HTML. **`<label>`** Представляет собой подпись к элементу пользовательского интерфейса. **`<legend>`** Представляет собой заголовок содержания родительского элемента `<fieldset>`. **`<meter>`**

Представляет собой скалярное значение в пределах известного диапазона или дробного значения. `<optgroup>` Позволяет группировать опции, находящиеся внутри элемента `<select>`. `<option>` Используется для определения пункта списка контейнера `<select>`, элемента `<optgroup>`, или элемента `<datalist>`. Элемент `<option>` может представлять раздел меню всплывающих окон и других перечней или списков HTML документа. `<output>` Является контейнерным элементом, в котором сайт или приложение могут выводить результаты вычислений или действий пользователя. `<progress>` Отображает индикатор, показывающий ход выполнения задачи, обычно отображаемый в виде прогресс бара (индикатора выполнения). `<select>` Представляет собой элемент управления который содержит меню опций: `<textarea>` (en-US) Represents a multi-line plain-text editing control, useful when you want to allow users to enter a sizeable amount of free-form text, for example a comment on a review or feedback form. Интерактивные элементы HTML содержит в себе элементы, которые помогают создавать интерактивные объекты пользовательского интерфейса.

Элемент Описание `<details>` Используется для раскрытия скрытой (дополнительной) информации. `<dialog>` Определяет диалоговое окно или другой интерактивный элемент, такой как инспектор или окно. Элементы `<form>` могут интегрироваться с диалогом с помощью указания атрибута `method=«dialog»`. Когда отправляется такая форма, диалог закрывается с `returnValue` равным `value` нажатой кнопки `submit`. `<summary>` The HTML Disclosure Summary element (`<summary>`) element specifies a summary, caption, or legend for a `<details>` element's disclosure box. Clicking the `<summary>` element toggles the state of the parent `<details>` element open and closed. Веб-компоненты Веб-компоненты - технология, относящаяся к HTML и делающая возможным создание и использование пользовательских элементов, как будто они в обычном HTML. Кроме того, вы можете даже создавать свои версии стандарта HTML элементов.

Примечание: Эти элементы определены в World Wide Web Consortium (W3C) Web Components collection of specifications скорее, чем в HTML спецификации. К тому же, спецификация Веб-компонентов не завершена и является темой для обсуждений.

Элемент Описание `<slot>` Является частью набора технологии Web Components, является заполнителем внутри веб компонента, который можно заполнить собственной разметкой, которая позволяет создавать отдельные деревья DOM и представлять их вместе. `<template>` Представляет собой механизм для отложенного создания клиентского контента, который не отображается во время загрузки страницы, но может быть инициализирован при помощи JavaScript. Устаревшие и осуждаемые элементы Предупреждение: Эти HTML-элементы являются устаревшими, и их использование не рекомендуется. Вы не должны использовать их для новых проектов, а также должны заменить их в старых проектах как можно быстрее. Они перечислены здесь только в информационных целях.

Элемент Описание `<acronym>` Позволяет явно указать, что данная последовательность букв, являющуюся акронимом или аббревиатурой. `<applet>` Определяет включение апплета Java. `<bgsound>` (en-US) Sets up a sound file to play in the background while the page is used; use `<audio>` instead. `<big>` (en-US) Renders the enclosed text at a font size one level larger than the surrounding text (medium becomes large, for example). The size is capped at the browser's maximum permitted font size. `<blink>` Causes the enclosed text to flash slowly. `<center>` (en-US) Displays its block-level or inline contents centered horizontally within its containing element. `<content>` (en-US) An obsolete part of the Web Components suite of technologies—was used inside of Shadow DOM (en-US) as an insertion point, and wasn't meant to be used in ordinary HTML. It has now been replaced by the `<slot>` element, which creates a point in the DOM at which a shadow DOM can be inserted. `<dir>` (en-US) Container for a directory of files and/or folders, potentially with styles and icons applied by the user agent. Do not use this obsolete element; instead, you should use the `` element for lists, including lists of files. `` Определяет размер шрифта, цвета и лицо для его содержимого

`<frame>` (en-US) Defines a particular area in which another HTML document can be displayed. A frame should be used within a `<frameset>` (en-US). `<frameset>` (en-US) Used to contain `<frame>` (en-US) elements. `<image>` > Примечание: Этот элемент был удалён из HTML5 (W3C) спецификации, но до сих пор остаётся в спецификации WHATWG. Он частично встроен в большинство браузеров, хотя бы поэтому вряд ли уйдёт. Поскольку схематический алгоритм не реализован ни в одном браузере, семантика тега `<hgroup>` на практике реализована только теоретически. Спецификация HTML5 (W3C) даёт совет как обозначать подзаголовки, альтернативные заголовки и слоганы. `<keygen>` Элемент HTML `<image>` был экспериментальным элементом, предназначенный для отображения изображений. Он никогда не был реализован, вместо него должен использоваться стандартный элемент ``. `<marquee>` Used to insert a scrolling area of text. You can control what happens when the text reaches the edges of its content area using its attributes. `<menuitem>` (en-US) HTML-элемент `<marquee>` используется для создания на странице прокручивающегося текста (бегущей строки). `<nobr>` (en-US) Prevents the text it contains from automatically wrapping across multiple lines, potentially resulting in the user having to scroll horizontally to see the entire width of the text. `<noembed>` (en-US) An obsolete, non-standard way to provide alternative, or «fallback», content for browsers that do not support the embed element or do not support the type of embedded content an author wishes to use. This element was deprecated in HTML 4.01 and above in favor of placing fallback content between the opening and closing tags of an `<object>` (en-US) element. `<noframes>` (en-US) Provides content to be presented in browsers that don't support (or have disabled support for) the `<frame>` (en-US) element. Although most commonly-used browsers support frames, there are exceptions, including certain special-use browsers including some mobile browsers, as well as text-mode browsers. `<param>` (en-US) Defines parameters for an `<object>` (en-US) element. `<plaintext>` (en-US) Renders everything following the start tag as raw text, ignoring any following HTML. There is no closing tag, since everything after it is considered raw text. `<rb>` (en-US) Used to delimit the base text component of a ruby annotation, i.e. the text that is being annotated. One `<rb>` element should wrap each separate atomic segment of the base text. `<rtc>` (en-US) Embraces semantic annotations of characters presented in a ruby of `<rb>` (en-US) elements used inside of `<ruby>` element. `<rb>` (en-US) elements can have both pronunciation (`<rt>` (en-US)) and semantic (`<rtc>` (en-US)) annotations. `<shadow>` An obsolete part of the Web Components technology suite that was intended to be used as a shadow DOM insertion point. You might have used it if you have created multiple shadow roots under a shadow host. `<spacer>` Allows insertion of empty spaces on pages. It was devised by Netscape to accomplish the same effect as a single-pixel layout image, which was something web designers used to use to add white spaces to web pages without actually using an image. However, `<spacer>` is no longer supported by any major browser and the same effects can now be achieved using simple CSS. `<strike>` (en-US) Places a strikethrough (horizontal line) over text. `<tt>` (en-US) Creates inline text which is presented using the user agent default monospace font face. This element was created for the purpose of rendering text as it would be displayed on a fixed-width display such as a teletype, text-only screen, or line printer. `<xmp>` (en-US) Renders text between the start and end tags without interpreting the HTML in between and using a monospaced font. The HTML2 specification recommended that it should be rendered wide enough to allow 80 characters per line.

From:
<http://git.wwooss.ru/> - **worldwide open-source software**

Permanent link:
http://git.wwooss.ru/doku.php?id=software:development:ps_pycharm:html5:element&rev=1691756342

Last update: **2023/08/11 15:19**

