

# Основы HTML

**HTML (Hypertext Markup Language)** - это код, который используется для структурирования и отображения веб-страницы и её контента. Например, контент может быть структурирован внутри множества параграфов, маркированных списков или с использованием изображений и таблиц данных. Как видно из названия, эта статья даст вам базовое понимание HTML и его функций.

## Что такое HTML на самом деле?

HTML не является языком программирования; это язык разметки, и используется, чтобы сообщать вашему браузеру, как отображать веб-страницы, которые вы посещаете. Он может быть сложным или простым, в зависимости от того, как хочет веб-дизайнер. HTML состоит из ряда **элементов**, которые вы используете, чтобы вкладывать или оборачивать различные части контента, чтобы заставить контент отображаться или действовать определённым образом. Ограждающие теги могут сделать слово или изображение ссылкой на что-то ещё, могут сделать слова курсивом, сделать шрифт больше или меньше и так далее. Например, возьмём следующую строку контента:

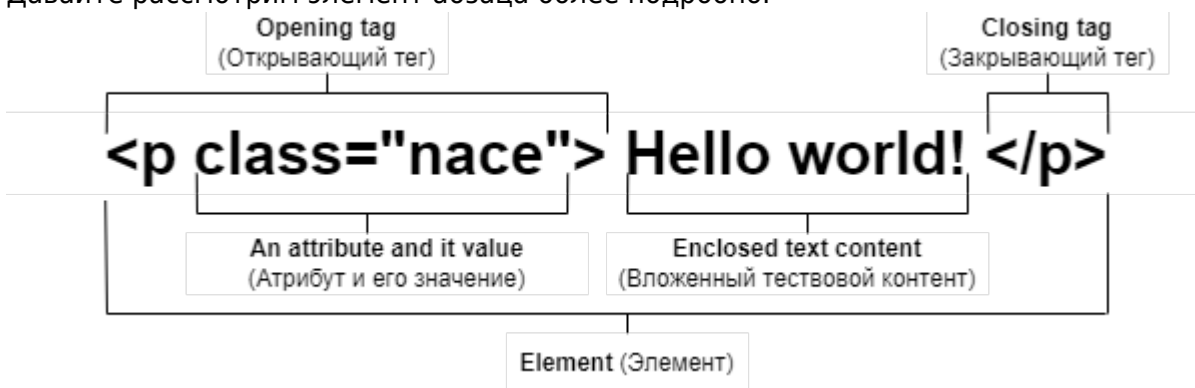
```
Моя кошка очень раздражена
```

Если бы мы хотели, чтобы строка стояла сама по себе, мы могли бы указать, что это абзац, заключая его в теги абзаца:

```
<p>Моя кошка очень раздражена</p>
```

## Анатомия HTML элемента

Давайте рассмотрим элемент абзаца более подробно.



## Ключевые особенности стандарта XHTML:

- все элементы пишутся строго в нижнем регистре;
- закрываются даже одиночные теги (например, `<img />`);

- основная кодировка – **UTF-8**;
- все атрибуты записываются только в развернутом виде (например, если в **HTML5** отключить возможность выбора элемента из выпадающего списка можно указанием простого атрибута **disabled**, то в **XHTML** обязательно писать **disabled=«disabled»**).

Верстать код сайта на основании стандарта **XHTML** не обязательно, так как **HTML5** и современные браузеры не столь требовательны, но он может встретиться (и этого не нужно пугаться).

## Консорциум Всемирной сети

Количество сайтов в сети настолько велико, что уже не хватает заложенного объема **IPv4-адресов** для их идентификации. С каждым годом число растет, что логично приводит к необходимости стандартизации их написания и отображения конечным пользователям.

Некоммерческая организация **W3-Консорциум** ([www.w3.org](http://www.w3.org)) разрабатывает единые принципы и рекомендации для производителей Интернет-контента, чтобы обеспечить максимальную совместимость между продуктами, платформами, программами. Это позволяет делать Всемирную сеть максимально удобной и универсальной.



Помимо непосредственно стандартов **HTML** компания занимается утверждением и других **связанных положений**:

1. CSS (каскадные таблицы стилей),
2. DOM (объектная модель документа),
3. PNG (формат хранения растровых изображений),
4. HTTP (протокол передачи данных),
5. RDF (модель представления метаданных),
6. XPath (язык для доступа к частям XML-документов) и др

Важно и то, что вы всегда можете проверить соответствие вашего кода стандарту при помощи

**валидаторов.** В сети их огромное множество и для разных целей: **HTML, CSS, JavaScript.** Сервисы позволяют валидировать свой или чужой код. Для этого требуется ввести в поисковике требуемый (например, **HTML5** валидатор), вставить туда код и произвести проверку. Все нарушения будут подсвечены соответствующим цветом и подсказками. Ими пользуются все разработчики для верификации качества работы программистов. Все современные **IDE** дают возможность валидировать код (в большинстве случаев для этого потребуется установка плагина).

## HTML теги. Структура тега



**Тег** – отдельный HTML элемент, позволяющий вкладывать или оборачивать части документа для определенного отображения или поведения.

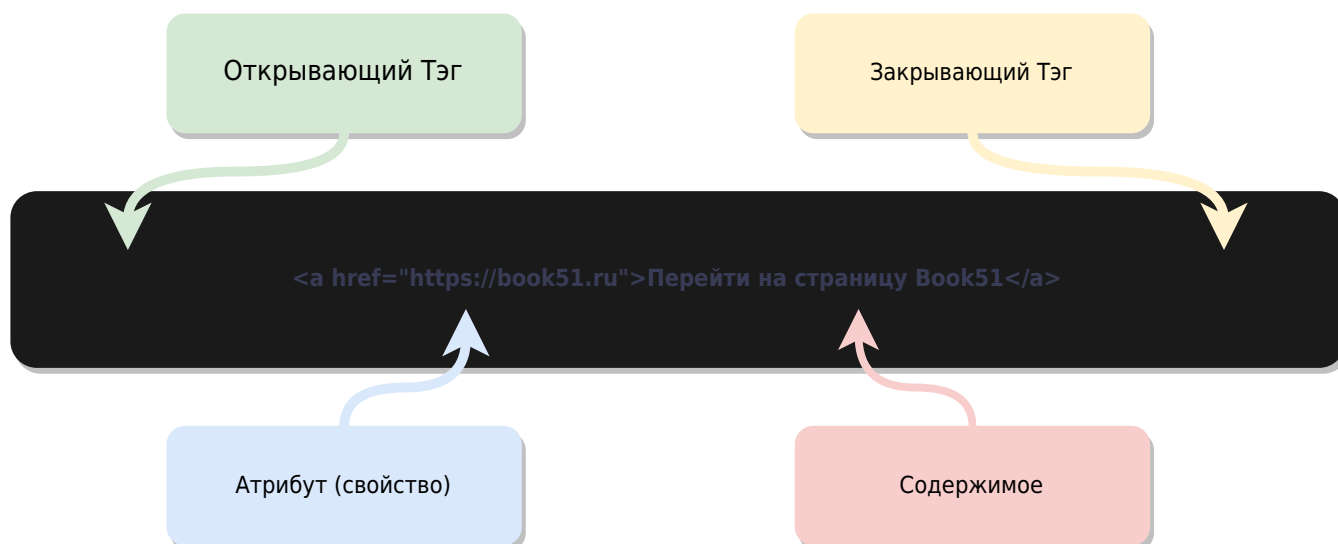
Говоря проще, при помощи тегов вы можете выделять текст, вставлять рисунки и видео, передавать служебную информацию, обозначать ссылки.

Теги могут дополняться атрибутами, свойствами. С их помощью можно модифицировать содержимое элемента, уникализировать его.

На текущий момент насчитывается около **120 тегов**, а если убрать устаревшие, то и того меньше. Их не так сложно запомнить, да и наиболее часто используемых несколько десятков.

Рассмотрим **типичную структуру web-элемента**:

1. **Открывающий тег** (все теги представляются в угловых скобках, в открывающем, если требуется, перечисляются атрибуты);
2. **Закрывающий тег** (присутствует не во всех тегах, идентифицируется косой чертой);
3. **Атрибут**, свойство (название свойства и его значение предпочтительно в двойных кавычках, возможно использовать и одинарные);
4. **Содержимое** (внутреннее содержимое тега, обычно в виде текста либо других вложенных тегов).



Подобным образом выглядит любой элемент web-страницы. При наличии нескольких атрибутов

они разделяются пробелом. Теги не всегда могут иметь свойства или содержимое.

Первичное знакомство с разнообразием html-элементов можно осуществить на любом сайте. Если вы находитесь в браузере Chrome, то прямо сейчас можно нажать сочетание клавиш **CTRL + U** (откроются **Инструменты разработчика**) и просмотреть код данной страницы. Фактически, весь этот набор тегов и свойств в сочетании с **JavaScript** обрабатывается браузером, чтобы вы смогли удобно пользоваться текущим сайтом.

Сайт представляет собой совокупность таких страниц, связанных между собой внутренними и внешними гиперссылками.

Далее поговорим о **классификации тегов и атрибутов**. Более детальное их описание и примеры будут представлены в следующем уроке.

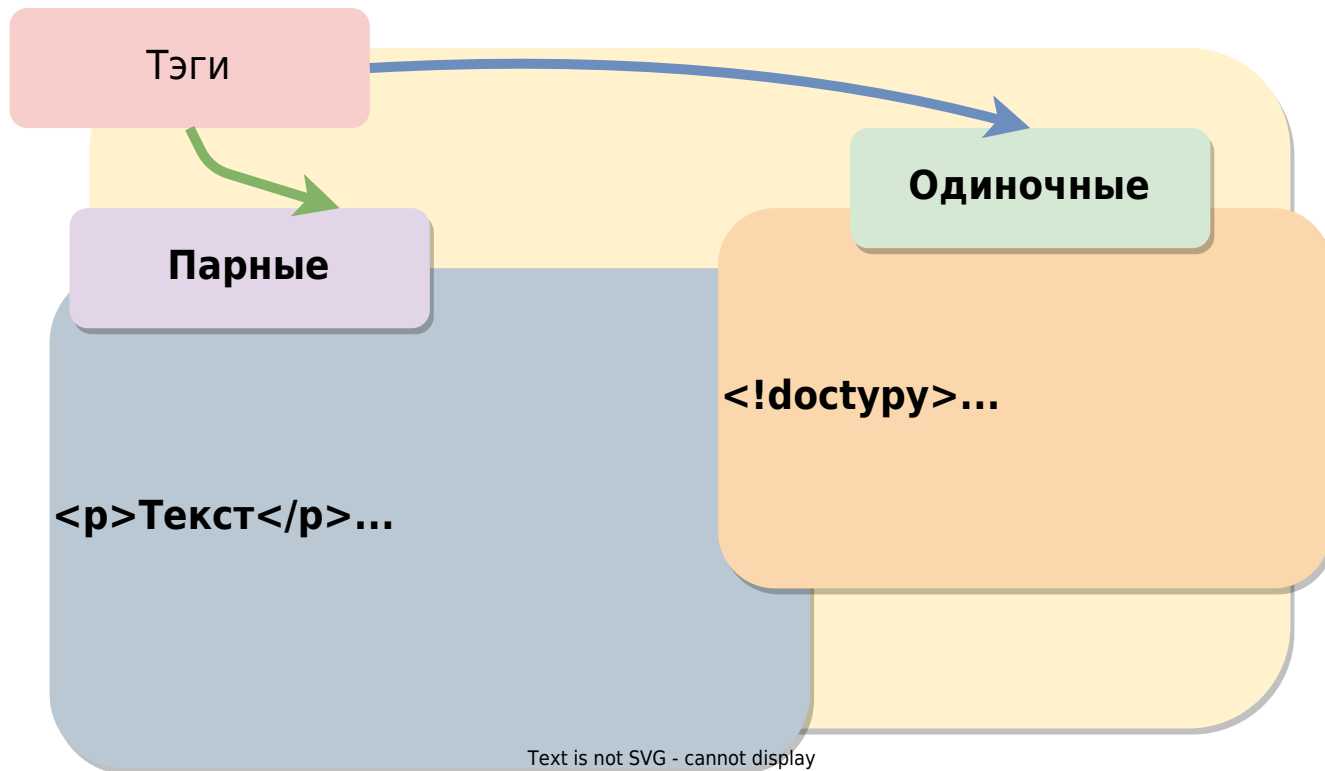
## HTML теги. Парные и одиночные теги

Как мы уже поняли выше, теги бывают парными или одиночными.

### Одиночные теги

Одиночных элементов насчитывается **16 штук**. Наиболее часто используемые:

1. **<!doctype>** Некая инструкция браузерам для определения того, к какому типу относится документ, какую версию **HTML** использовать.
2. **<meta>** Необходим для задания дополнительной информации о странице, указывается в заголовке **HTML-документа**. Как и вышеупомянутый тег, не влияет на внешний вид страницы, так как носит служебный характер.
3. **<hr>** Задаёт горизонтальную линию для визуального отделения элементов страницы, применяется для отделения заголовка от основного текста, выделения логических блоков. По умолчанию представляет собой серую линию толщиной в 1 пиксель.
4. **<br>** Перенос нижеидущего содержимого на новую строку. Может потребоваться для избегания длинных полотён текста. Небольшие куски информации воспринимаются лучше, чем огромный абзац.
5. **<img>** Чтобы вставить на страницу графический элемент. Рисунки можно продемонстрировать в одном из следующих форматов: **jpg, png, gif**. Само изображение в документ не вставляется непосредственно, указывается ссылка на него (локальная или из другого источника), а уже браузер загружает картинку и демонстрирует пользователю.
6. **<input>** Применяется в формах и задаёт тип полей (могут быть текстовыми, числовыми, позволяют выбрать дату, файл, отмечать те или иные элементы списка и т.п.).



## Парные тэги

Парные тэги обязательно имеют **закрывающий тег** и содержимое (пусть даже и пустое). В **HTML** их больше всего (около **100**). В качестве примера приведем некоторые:

1. **<p>Текст</p>** Представляет заключенный внутри текст в виде отдельного абзаца. По умолчанию он имеет отступ и отделяется от предыдущего и последующего элементов дополнительным разрывом (это поведение можно поменять).
2. **<a href=«#»>Текст ссылки</a>** Используется для задания и представления в документе ссылки (она может вести на другую страницу сайта, определенный участок на самой странице или любой адрес в Интернете).
3. **<small>Текст</small>** Делает текст меньшего масштаба по сравнению с основным.
4. **<mark>Текст</mark>** Выделяет текст по подобию маркера (аналог того, как на листах бумаги подчеркивают самое важное фломастером) желтым цветом.



С помощью CSS имеется возможность менять поведение практически всех тегов.

## HTML тэги. Блочные и строчные тэги

Поведение любого тега сегодня (в стандарте **HTML5**) легко изменить по своему усмотрению. Деление элементов на блочные и строчные сложилось исторически (именно поэтому они изначально относятся к одному из этих типов). Сейчас таких типов больше, но их определение связано с каскадными таблицами стилей, о которых пойдет детальная речь в одном из последующих уроков.

**Блочные теги** занимают всю ширину страницы или родительского элемента. Если вы зададите два таких тега подряд, то каждый из них будет начинаться как бы с новой строки, абзаца.

**Строчные теги** не имеют строго размера. Он зависит от того, сколько символов в них содержится. Несколько подряд идущих строчных элементов не будут разделяться новыми строками, а будут выстраиваться друг за другом.

В **CSS-свойствах** отображение тегов можно менять по следующим типам:

- 
1. **Блочные (display: block)** – получает свойство блочности, занимает всю ширину страницы;
  2. **Строчные (display: inline)** – тег становится строчным;
  3. **Флекс (display: flex)** – элемент приобретает свойство особого типа блоков – флекс (ведет себя как блок, но внутреннее содержимое гибко настраивается);
  4. **Грид (display: grid)** – еще один блочный тип с особенностями внутреннего поведения элементов (строятся на основании «сетки», состоящей из строк и колонок);
  5. **Таблица (display: table)** – тег и его содержимое будут наследовать свойства таблиц;
  6. **Строчный блок (display: inline-block)** – блочный элемент ведет себя как строка, но сохраняет часть свойств (можно задать размер, границы).
  7. **Спрятанный (display: none)** – делает элемент невидимым и полностью удаляет его отображение со страницы.

Существуют и другие свойства отображения, многие из которых узконаправлены и зачастую экспериментальны.

## 1. Блочный тэг

**Самый популярный блочный тег времен HTML 4. Основная его задача - отобразить содержимое как блок. Он и сегодня входит в список часто используемых. Достаточно открыть код любой страницы в сети и проанализировать его. Почти везде**

применяется максимально широко. Посмотрим на простой пример.

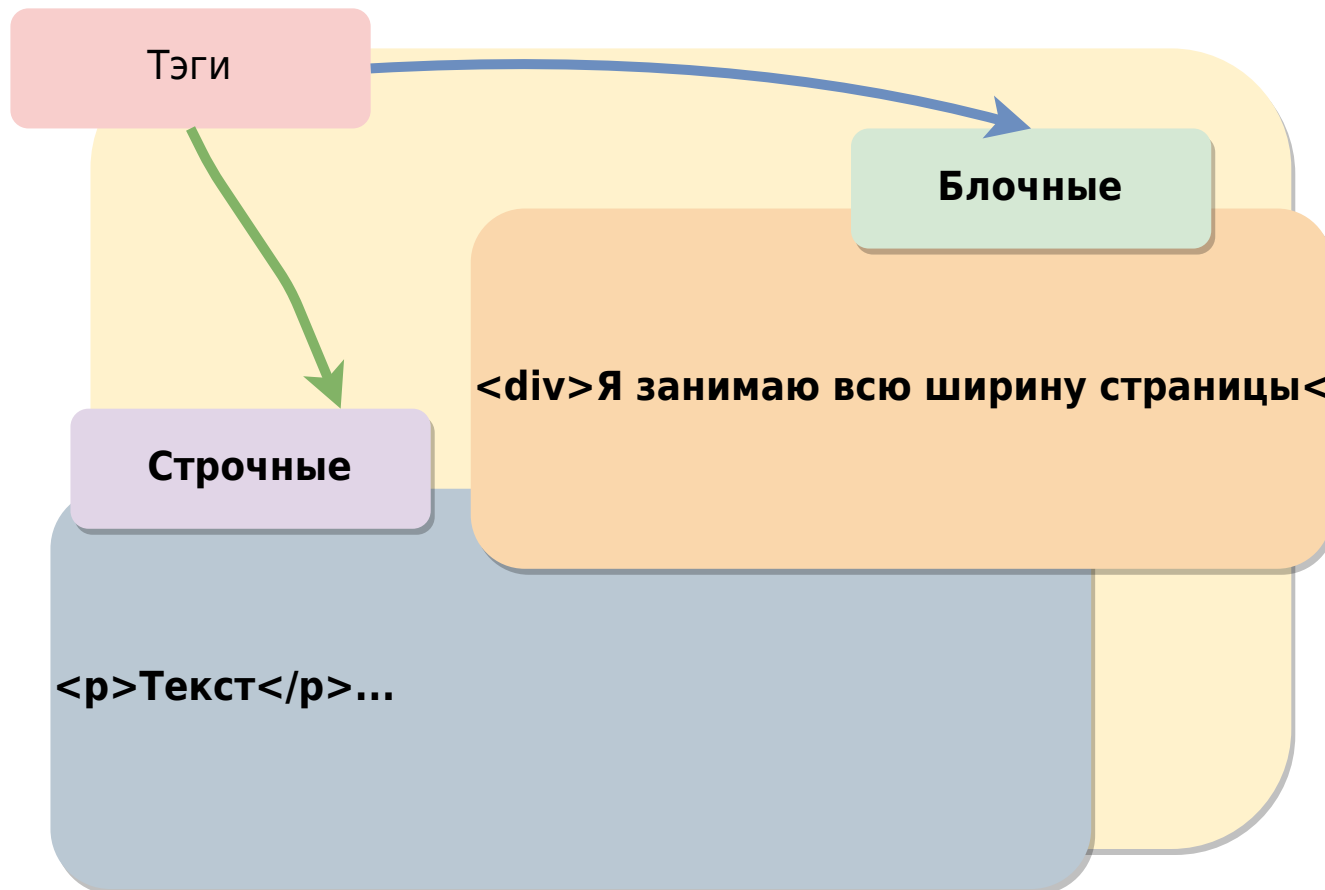
[Пример - HTML](#)

```
<div>Этот текст занимает всю ширину страницы</div>  
<div>Этот текст начнется с новой строки</div>
```

Если сохранить приведенный код в виде **html-файла** и открыть в браузере, то увидим, что куски текста начинаются с разных строк. Причина – в сути блочных элементов (они стремятся занять всю доступную ширину, которая им предоставлена).

Другие блочные теги:

1. **<table></table>** Позволяет рисовать таблицы
2. **<p>Параграф</p>** Обозначает параграф текста
3. **<h1>Большой заголовок</h1>** Для оформления заголовка самого верхнего уровня (в роли такого обычно выбирается основная статья страницы).
4. **<form></form>** Отражает на сайте форму (авторизации, регистрации, опроса и др.)



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

Permanent link: [http://git.wvoss.ru/doku.php?id=software:development:ps\\_pycharm:html5:lesson:lesson.html\\_1&rev=1692548980](http://git.wvoss.ru/doku.php?id=software:development:ps_pycharm:html5:lesson:lesson.html_1&rev=1692548980)

Last update: 2023/08/20 19:29

