**[HTML для начинающих](https://www.youtube.com/playlist?list=PLY4rE9dstrJyeZlPWoKJr1xKVVnG4w-Hc)**

Содержание:

[1**.** HTML для начинающих – **#1** – Введение………………………………………………………………………………………………… 1](#Урок_1)

[2**.** HTML для начинающих – **#2** – Теги и атрибуты……………………………………………………………………………… 2](#Урок_2)

[3**.** HTML для начинающих – **#3** – Списки……………………………………………………………………………………………………… 5](#Урок_3)

[4**.** HTML для начинающих – **#4** – Таблицы…………………………………………………………………………………………………… 7](#Урок_4)

[5**.** HTML для начинающих – **#5** – Фрэймы……………………………………………………………………………………………………… 8](#Урок_5)

[6**.** HTML для начинающих – **#6** – Ссылки…………………………………………………………………………………………………… 10](#Урок_6)

[7**.** HTML для начинающих – **#7** – Атрибуты**.** Часть 1……………………………………………………………………… 10](#Урок_7)

[8**.** HTML для начинающих – **#8** – Атрибуты**.** Часть 2……………………………………………………………………… 12](#Урок_8)

[9**.** HTML для начинающих – **#9** – Формы**.** Введение…………………………………………………………………………… 15](#Урок_9)

[10**.** HTML для начинающих – **#10** – Формы**.** Часть 2………………………………………………………………………… 16](#Урок_10)

[11**.** HTML для начинающих – **#11** – Семантические особенности HTML5…………………………… 18](#Урок_11)

**[1. HTML для начинающих – #1 – Введение](#_top)**

Для работы с HTML лучше всего использовать текстовые редакторы:

1. **notepad++** – имеет подсветку синтаксисов и очень много плагинов и он гибкий. Он только под Windows!!!

2. **Subleme Text** – имеет все те же преимущества что и notepad++ плюс импортируется под все ОС, но он платный!!!

3. Браузер Chrome или FireFox.

4. Источник информации <htmlbook.ru>

HTML – Hipper text markup language – язык гипертекстовой разметки.

Любая веб страница состоит из основных частей:

Скелет любой HTML страницы:

<html> **-------------------------|**

<head> **----|- контейнер head |**

</head> **----| |**

**| – Контейнер html**

<body> **----|- контейнер body |**

</body> **----| |**

**|**

</html> **--------------------------|**

где <html></html>– контейнер в котором содержится все остальное содержимое страницы!!!

**Контейнер** – это парный тег, имеющий открывающую <html> и закрывающую </html> части!!!

Весь полезный контент находится между двумя элементами <html>и</html>.

<head></head> – находится всегда в верхней части страницы, который содержит в себе служебные элементы!!! – служебные теги, мета теги предназначенные для работы с поисковыми системами и браузерами!!!

Содержимое тега <head></head> НЕ ВИДНО обычному пользователю!!!

Пользователь видит только то, что находится в теге:

<body></body> – это основное тело страницы, которое содержит рабочие элементы!!!

Сейчас эра **Web 2.0** – теперешняя спецификация HTML уже пятая – HTML5 позволяет создавать не только сайты, а веб приложения!!!

Скелет любой HTML страницы имеет вид:

<!doctype html>

<html lang=**"en"**>

<head>

<meta charset=**"UTF-8"**>

<title>**Document**</title>

</head>

<body>

**...**

</body>

</html>

Подробнее:

1. Тег <!doctype html> – применяется для того чтобы браузер понимал с какой конкретной версией HTML страницы он на данный момент работает!!!

2. В теге <head></head> – всегда должны быть мета-теги!!!

2а. Мета-тег charset представляет из себя мета-тег, который указывает браузеру в какой кодировке символы расположены на странице!!!

2б. Второй очень важный элемент – это <title></title> – это название самого документа, то как будет выглядеть вкладка страницы!!!

3. Тег <body></body> – содержит в себе:

Тег <H1></H1> – заголовок Веб страницы

Тег <p></p> – тег абзаца!!!

**[HTML для начинающих – #2 – Теги и атрибуты](#_top)**

Все теги в зависимости от свойств и области применений разделены на группы по типам!!!

Условно теги делятся на следующие типы:

1. **Теги верхнего уровня**.

К ним относятся контейнеры:

<html></html> – содержит весь документ html;

<head></head> – голова документа html;

<body></body> – тело документа html.

2. **Теги заголовка документа**.

К ним относятся контейнеры:

<title></title> – заголовок документа

<meta></meta> – тег вмещающий в себя информацию, необходимую для работы браузера и поисковых систем.

3. **Блочные элементы** – служат на странице для указания размещения элементов страницы:

Например:

<H1>**Тег размещения заголовка текста**</H1>

Различаются 6 уровней заголовка текста – <H1></H1>, <H2></H2>, <H3></H3>, <H4></H4>, <H5></H5>, <H6></H6>:

<H1> – самый большой!!!

<H6> – самый маленький!!!

Тег <DIV></DIV> представляет из себя универсальный блочный элемент!!!

В него включены теги:

<P></P> – тег абзаца!!!

<HR> – тег, который рисует горизонтальную черту на странице!!!!

<PRE></PRE> – тег предварительного форматирования!!!

Без этих тегов текст:

Шалтай-Болтай

Сидел на стене.

Шалтай-Болтай

Свалился во сне.

Будет иметь вид одной строки:

Шалтай-Болтай Сидел на стене. Шалтай-Болтай Свалился во сне.

!Если заключить этот текст в тег предварительного форматирования <PRE></PRE> получим:

Шалтай-Болтай

Сидел на стене.

Шалтай-Болтай

Свалился во сне.

!Что записали то и вывелось!!!

Или каждую строчку можно заключить в тег абзац:

<p>**Шалтай-Болтай**</p>

<p>**Сидел на стене.**</p>

<p>**Шалтай-Болтай**</p>

<p>**Свалился во сне.**</p>

!Но между строчками увеличиться интервал!!!

4. **Строчные элементы** – служат не для изменения расположения текста на странице, а для изменения его свойств и внешнего вида.

Если заключить текст:

<p>**Шалтай-Болтай**

**Сидел на стене.**

**Шалтай-Болтай**

**Свалился во сне.**</p>

В один тег абзаца <p></p> то снова получиться одна строка:

Шалтай-Болтай Сидел на стене. Шалтай-Болтай Свалился во сне.

Но если в конце каждой строчки добавить в тег <br> то он перенесет каждую строчку на новую!!!

Тег <br> переносит текст на другую строчку но уже без отступа!!! В отличии от тега <p></p>!!!

<p>**Шалтай-Болтай**<br>

**Сидел на стене.**<br>

**Шалтай-Болтай**<br>

**Свалился во сне.**</p>

Получим:

Шалтай-Болтай

Сидел на стене.

Шалтай-Болтай

Свалился во сне.

Тег <b></b> используется для того, чтобы выделять текст жирным!!!!

Тег <em></em> – курсив!!!

Тег <big></big> – делает строчку больше!

Тег <small></small> – делает строчку меньше!

Различие строчных и блочных элементов:

**Блочные элементы**:

- занимают всю доступную ширину!!!

- высота зависит от содержимого;

- каждый новый элемент занимает новую строку;

- могут вкладываться в другие блочные элементы;

- не могут быть вложены в строчные элементы!!!!

**Строчные элементы**:

- ширина зависит от содержимого + поля и отступы;

- высота зависит от содержимого;

- могут занимать одну строку;

- могут вкладываться в блочные и строчные элементы!!!

Тег <ul></ul> – неупорядоченный список

Теги <li></li> – элементы неупорядоченного списка <ul></ul>.

Тег <sup></sup> – перемещает элемент в верхний регистр, например обозначение степени.

Тег <img src=**""** alt=**""**>используется для вставления картинки на страницу!!!!

Атрибуты данного тега:

src=**""** – от английского source – это путь до картинки!!!

alt=**""** – используется для показа альтернативного текста если картинку не возможно будет отобразить!!!

**Атрибут** – это элемент тега, который служит для изменения его свойств либо добавления дополнительных свойств!!!

У тега <body></body>

Атрибут style данного тега может изменить цвет страницы:

<body style = **"background-color: #DF8C5D"**>

где **#DF8C5D** – название цвета в 16-тиричном исчислении.

4. Универсальные элементы;

5. Списки;

6. Таблицы;

7. Фреймы;

8. Семантическая разметка.

**[HTML для начинающих – #3 – Списки](#_top)**

**Списки** – это то что мы пишем для того, чтобы как-то структурировать информацию.

Списки бывают: списки книг, списки музыкальных групп, списки тегов html.

**Список** – это перечень пунктов!!!

Виды списков:

1. **Маркированный** – у такого списка перед каждым пунктом имеется так называемый маркер, который представляет собой закрашенный кружок "●"

Например:

● первый пункт

● второй пункт

● третий пункт

Контейнером этого списка является тег <ul></ul> – **неупорядоченный список**.

<ul> – это **unordered list** неупорядоченный список!!!

А пункты обрамляются тегами <li></li> – элементы неупорядоченного списка <ul></ul>.

<li> – это list item – элемент списка﻿!!!

Для задания внешнего вида маркеров предусмотрен атрибут type, который для тега <ul></ul> принимает следующие значения:

- type = **"disk"** – маркер в виде закрашенного круга (по умолчанию);

- type = **"circle"** – маркер в виде не закрашенного круга;

- type = **"square"** – маркер в виде закрашенного квадрата;

2. **Нумерованный** – то есть пункты обозначаются цифрами!!! Этот список обозначается тегом <ol></ol>, а пункты также обрамляются тегами <li></li>:

1. первый пункт

2. второй пункт

3. третий пункт

<ol> – это **ordered list** – упорядоченный список!!!

<li> – это **list item** – элемент списка﻿!!!

!Если требуются другие варианты нумерации то применяется снова атрибут type:

- type = **"1"** – нумерация по умолчанию.

- type = **"A"** – пункты обозначаются прописными латинскими буквами;

- type = **"a"** – то же самое, но строчное начертание;

- type = **"I"** – нумерация римскими цифрами в верхнем регистре;

- type = **"i"** – то же самое но в нижнем регистре.

!Если начало счета нужно задать не с единицы то применяется атрибут start, например:

start = **"2"**

3. **Список определений** – этот вид списков сложнее потому что он состоит не просто из отдельных пунктов, но также к каждому пункту дается соответствующее определение.

Он представляет из себя структуру следующего вида:

**первый пункт**

**описание первого пункта**

**второй пункт**

**описание второго пункта**

**третий пункт**

**описание третьего пункта**

Этот список состоит не из двух видов тегов, а из трех!!!

1. Это тег контейнер – <dl></dl>

2. Тег пунктов – <dt></dt>

3. Тег определения пунктов – <dd></dd>

Применение этого вида списков бывает удобным в некоторых специфических случаях например для анекдота с диалогами или для описания свойств элементов html.

Любой список образуется двумя видами тегов:

1. Общий контейнер списка

2. Контейнер разделения пунктов

**[HTML для начинающих – #4 – Таблицы](#_top)**

Таблицы в html создаются с помощью трех видов тегов:

1. Контейнер <table></table> – который представляет из себя тело таблицы!!!

Этот контейнер состоит из двух сущностей – это:

1. Строки – <tr></tr>;

2. Ячейки – <td></td>.

Например, таблица в HTML имеет вид:

<table>

<tr>

<td></td>

</tr>

</table>

Основные атрибуты таблицы:

- border – отвечает за толщину рамки. border = **"2"** – толщина рамки **2 pixels** (**2** пикселя)!!!!

- bordercolor – отвечает за цвет рамки. Рамка обычно рисуется как трехмерная, но при добавлении этого атрибута, становится однотонной.

- cellspasing – удаление отступов между ячейками.

- bgcolor – задает цвет фона.

- cellpadding – задает расстояние между контентом и рамкой ячейки.

<table cellspacing=**"0"**> – удаляет расстояние между таблицей и ячейками!!!

- align – выравнивание содержимого ячейки.

<table align = **"center"**> – выравнивание таблицы по центру!!!

Можно применять к ячейкам:

<td align = **"center"**>**Ячейка**</td>– выравнивание одной ячейки по центру!!!

И ко всей строке:

<tr align = **"center"**> – выравнивание всей строки по центру!!!

- width – задает ширину таблицы в процентах либо в пикселях!!!.

<table width=**"80%"**>– увеличение ширины таблицы на 80% всего экрана.

Нет необходимости запоминать все параметры атрибутов таблицы потому как **CSS** – Cascading Style Sheets – Каскадная таблица стилей, в которой все эти параметры прописываются более гибко и дают более красивый результат!!!

Объединение ячеек таблицы происходит с помощью двух свойств:

- colspan – объединение ячеек по горизонтали;

<td colspan=**"3"**></td>– объединение трех ячеек в одну!!!

- rowspan – объединение ячеек по вертикали.

<td rowspan=**"2"**></td> – объединение двух ячеек в одну!

Объединять можно любое количество ячеек как по горизонтали так и по вертикали!!!

Но есть ограничение – максимально допустимое значение – **65 534 ячейки**!!! Больше объединять нельзя!!!

!Таблицы еще можно влаживать одна в другую!!!

Таблицы предназначены для структурирования данных!!!

Раньше для верстки сайтов применялись таблицы!!! Но сейчас нет!!!

Почтовые рассылки верстаются с помощью таблиц, потому что по другому их не сверстать!!! Не позволяют почтовые службы!!!

Сайты с табличной версткой лучше не создавать!!!

**[HTML для начинающих – #5 – Фрэймы](#_top)**

**Фрэйм** – с английского "рамка", это особая выделенная область html документа, которая ссылается на другой документ и отображается этого документа.

Фреймы не получили широкого распространения!!! А сейчас они практически вытеснены технологиями CSS и JavaScript, которые обладают всеми достоинствами присущими фрейму и позволяют избежать их недостатков!!!

Эта технология применяется только в специфичных проектах.

С помощью фрейма веб-страницу можно разграничить Веб содержимое которой будет независимо друг от друга.

Так например в одну часть страницы можно поместить меню навигации сайта, а во втору содержимое сайта!!!

При этом загрузка содержимого осуществляется только на части сайта, что позволяет сократить объем передаваемой информации.

Для сознания простейшей фреймовой страницы достаточно три документа: основной и два внутренних!!!

Условно их можно назвать меню и контент.

Для создания документа распределенного на фреймы применяется немного другой скелет документа!!!

А именно контейнер <body></body> заменяется контейнером <frameset></frameset>!!!

У тега <frameset></frameset> независимо от желаемого расположения рамок присваивается один из двух атрибутов:

- cols – колонки;

- rows – строки;

В обоих случаях атрибуты задаются в пикселях и процентах по следующему синтаксису:

<frameset cols=**"ширина\_1, ширина\_2, ..."**>**...**</frameset>

Одна из величин может быть обозначенная звездочкой означающая остальное свободное место!!!

Можно задать любое количество фреймов, от этого ничего не будет разве что может немного увеличиться нагрузка на сервер и как следствие получить долгую загрузку. Лучше уж так не делать.

То же самое можно сделать с атрибутом rows только теперь вместо столбиков получаться строки:

<frameset rows=**"ширина\_1, ширина\_2, ..."**>**...**</frameset>

Отдельные документы создаются по стандартным правилам с использованием контейнера <body></body> и подключаются к главному документу с помощью одиночного тега <frame>!!!

С атрибутом src=**""** к которому в качестве значения присваивается путь до нужного документа. Подключать можно сторонние документы. Если в качестве значения атрибута src=**""** передать url=**""** адрес любого стороннего сайта, правда, у передаваемого сайта может быть защита от встраивания!!!

Также у тега <frame> есть такие атрибуты:

- src=**""** – путь до нужного документа.

- name=**""** – присваивает каждому отдельному фрейму имя, с помощью которого можно управлять параметрами фрейма.

- noresize – запрещает изменять размер фрейма.

- scrolling – управляет возможностью прокручивать содержимое фрейма.

Разделения страницы по фреймам **50%**

<!doctype html>

<html lang=**"en"**>

<head>

<meta charset=**"UTF-8"**>

<title>**Document**</title>

</head>

<frameset cols=**"50%, \*"**>

<frame src=**"**http://loftblog.ru/**"**>

<frame src=**"**http://loftblog.ru/**"**>

<frame src=**"**http://loftblog.ru/**"**>

</frameset>

</html>

Чтобы сделать гибридный вариант разделения по строкам и колонкам тогда в тег <frameset> нужно добавить тег <frameset> с тегом <frame> по следующей схеме:

<frameset>

**| |**

**|** <frame>

**|**

**|** <frameset>

**| | | |**

**| | |** <frame>

**| |** <frame>

**|** </frameset>

</frameset>

Код html:

<!DOCTYPE html>

<html lang=**"en"**>

<head>

<meta charset=**"UTF-8"**>

<title>**Rame2**</title>

</head>

<frameset rows=**"35%, 65%"** scrolling=**"no"**>

<frame src=**"Tabele.html"** noresize>

<frameset cols=**"840, \*"**>

<frame src=**"Liste.html"** scrolling=**"no"**>

<frame src=**"Reteta.html"** scrolling=**"no"**>

</frameset>

</frameset>

</html>

**[HTML для начинающих – #6 – Ссылки](#_top)**

Для создания ссылки нужен простой тег:

<a href = **"url"**></a>

где a – ссылка.

Основной атрибут href = **"url"**

**url** – **universal resurs locator** – это путь до того документа который мы хотим посетить, на который мы ссылаемся.

Атрибут target=**"\_blank"** используется для открытия ссылки в новой вкладке!!!

В зависимости от того где находится документ на который мы ссылаемся прописываются разные виды путей, по этим путям ссылки делятся на два вида:

1. **Абсолютные** – применяются если документ имеется на другом ресурсе!!!

В данном случае ссылка начинается с указания протокола **http://** и **адрес**!!!

2. **Относительные** – те, что расположены на том же ресурсе относительно исходного документа!!!

Если документ находится в папке то ссылку нужно указывать через слеш – **/**, например:

<a href=**"Newlesson6/Lesson 6\_2 – Links.html"** target=**"\_blank"**>**Перейти на следующую страницу**</a>

А обратную через **../** – значит выйти с папки!!!! и поискать файл за пределами папки:

<a href=**"../Lesson 6 – Links.html"**>**Перейти на предыдущую страницу**</a>

То есть если оба документа находятся в разных папках, но обе папки находятся в одной общей то перед названием директории пишется две точки и слеш **../**

Если документы разделены большим количеством папок – то количество точек и слешей соответственно увеличивается пропорционально количеству этих папок, разделяющих файлы!!!

**[HTML для начинающих – #7 – Атрибуты. Часть 1](#_top)**

**Атрибуты** – это некоторые параметры html тегов, которые позволяют сделать тот или иной элемент более уникальным!!!

То есть с помощью них можно указать браузеру как он должен формировать этот элемент!!!

Атрибутов для тегов можно добавлять неограниченное количество, при этом нужно их разделять пробелами!!!!

**Порядок атрибутов не важен!**

Сначала записывается имя атрибута потом знак равно и в кавычках значение атрибута:

<a href = **"/about.html"** title = **"О нас"** class = **"menu-link"**>**О нас**</a>

где <a></a> – тег

title = **"** **"** – атрибут

**"menu-link"** – значение.

Например – id=**""**, class=**""**, data-\*=**""**, style=**""**, title=**""**, alt=**""**, src=**""**, href=**""**, target=**""**, lang=**""**, charset=**""**, width=**""**, height=**""**, border=**""**, align=**""**, bordercolor=**""**, bgcolor=**""**, face=**""**, size=**""**, color=**""**...

Атрибутов существует великое множество но они не применяются потому как Веб разработка разделена на три независимые части:

1. Разметка – **HTML** оформление!!!

2. **CSS** – оформление;

3. **JavaScript** – функциональность!!!

Поэтому часть атрибутов просто не нужна так как их довольно много то запоминать их нет смысла!!!

Любой незнакомый атрибут можно найти на [**http://htmlbook.ru/**](http://htmlbook.ru/)

Однако среди всех атрибутов можно выделить атрибуты:

1. **Универсальные** – применяются практически ко всем тегам, поэтому выделены в отдельную группу, чтобы не повторять их для всех тегов.

2. **Специфические** – применяются не ко всем тегам.

Чтобы указать адрес любой совершенно неважной картинки можно использоваться сайтом <http://lorempoixel>

Чтобы указать произвольное количество слов нужно установить в Sublime text плагин Emmet и написать Lorem + количество нужных слов + нажать клавишу Tab, например

Lorem100+Tab -> получим текст:

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Officiis ut fuga molestiae laboriosam pariatur nam, sint possimus nulla exercitationem illum, voluptates, dicta ipsam inventore eveniet illo maiores expedita reiciendis quibusdam a. Porro possimus libero natus veritatis maiores, quidem placeat ea rem aliquam illo ut cumque vel sunt est magni nam voluptate molestias quasi explicabo tempore. Culpa inventore amet, ullam sequi, neque suscipit explicabo dolores illum delectus, nostrum, assumenda fuga ad sunt autem sapiente earum quidem a qui! Illo error, itaque eaque neque? Nulla, nam unde suscipit impedit mollitia, nobis perspiciatis magni aliquam dolores illo reiciendis consectetur dolor, quasi neque earum!

Или можно использовать <www.blindtextgenerator.com/> который генерирует требуемое количество слов!!!!

Сейчас сайты верстаются не с помощью таблиц, а применяется блочная верстка сайтов!!!

То есть тег <table></table> используется только для таблиц!!!

Чтобы обернуть что-то в тег – нужно выделить этот участок и нажать **Ctrl + Alt + Enter** и далее написать тот тег, который хочешь.

Код, в котором теги используются по своему прямому назначению и хорошо структурированы называется **семантичным**!!!

Его больше остальных любят поисковики и выставляют разработанный сайт на лучшие позиции!!!!

**[HTML для начинающих – #8 – Атрибуты. Часть 2](#_top)**

За счет атрибутов можно придать дополнительных особенностей документу html.

Большинство атрибутов форматирования уже не используются но знать их все же нужно чтобы понимать азы HTML.

Все устаревшие атрибуты – это в основном атрибуты форматирования текста и придания какого-то стиля – цвета, фона, рамок и т.д.

1. align = "" (выравнивание) – он как и в вордовском документе может принимать 4 значения!!!

То есть выравнивание текста по:

1. align = **"left"** – по левому краю

2. align = **"right"** – по правому краю

3. align = **"center"** – по середине

4. align = **"justify"** – по ширине

Атрибуты для текста:

Например – выравнивание по центру

<h1 align=**"center"**>**Мой сайт портфолию**</h1>

Добавление красной разделительной черты:

<hr color=**"red"**>

Атрибуты для изображений:

<img src=**"http://lorempixel.com/200/200/sports"** alt=**""**>

1. src=**""** – source – источник – путь к изображению, которое мы хотим вставить!!!

2. alt=**""** – устанавливает альтернативный текст для изображений!!! Этот текст позволяет получить альтернативный текст при отключенной загрузке изображений!

3. title=**""** – всплывающая подсказка при наведении курсора. Для поисковика этот атрибут позволяет лучше ранжировать содержимое сайта!!!

4. Размеры картинки:

a) width=**"200"** – ширина

b) height=**"200"** – высота

c) border=**"2"** – граница **2** пикселя!!!

Атрибут border=**"2"** чаще всего применяется в таблицах!!!

При указании не соответствующих размеров картинки она будет деформироваться!!!

Поэтому в значения width=**"200"** и height=**"200"** нужно указывать значения исходного изображения!!!

Для тега **<img** очень важен атрибут src=**""** так же как и для тега ссылки **<a** важен атрибут href=**""** так как именно в нем указывается адрес документа на который мы хотим перейти!!!

Для ссылок есть атрибут target=**""**, он служит для указания того где мы хотим открыть нашу ссылку:

target=**""**:

1. **\_parent** – нужен для работы с фреймами!

2. **\_top** – нужен для работы с фреймами!

Первые два атрибута используется достаточно редко!!!

3. **\_blank** – нужен для того, чтобы открыть ссылку в новой вкладке! Нужен, например, для пользователей – когда пользователю нужно заглянуть на страничку и не уходить с основного сайта!!!

4. **\_self** – открыть ссылку в этой же вкладке, именно он используется по умолчанию для всех ссылок!!!

Атрибут title используется для отображения всплывающей подсказки при наводе курсора мышки на ссылку!!!

Для таблиц всегда нужно пользоваться справочником атрибутов [**http://htmlbook.ru**](http://htmlbook.ru) все их изучать не стоит!!!

**Так как инженер не должен все знать – инженер должен знать где найти!!!**

Для таблиц атрибуты:

1. border = **"1"** – граница – **1** пиксель!!!

2. cellpadding = **"1"** – добавления внутренних полей размером **10** пикселей вокруг содержимого ячеек!!!

3. cellspacing = **"1"** – отступы между ячейками – **1** пиксель!!!

Атрибут bgcolor применим не только к таблице!!! С помощью него можно задать фон и для всей страницы!!!

Например для таблицы:

<tr bgcolor=**"#cacaca"**>

Для фона страницы:

<body bgcolor=**"#cfcfcf"**>

Разработчики давно уже пришли к пониманию того что нужно разделять между собой:

1. Структуру сайта – **HTML**;

2. Внешний вид – **CSS**;

3. Функционал – **JavaScript**.

То есть стараться использовать независимо друг от друга HTML, CSS и JavaScript.

Поэтому чтобы создать какие-то точки связи между этими трема технологиями и есть два специальных атрибута:

1. id=**""** – это уникальный идентификатор элемента на страничке. Он может быть присвоен любому тегу!!! Но его значение для каждого тега должно быть уникальным!!! То есть должно встречаться только один раз!!! Поэтому рекомендуется давать какие-то осмысленные имена для своего же удобства!!!

2. class=**""** – служит для связывания тегов с определенным стилевым оформлением! То есть если имеем несколько заголовков и их необходимо выделить по особенному – если к каждому заголовку мы будем применять специальный атрибут копировать его и вставлять по 100 раз!!! Потом вдруг захотим изменить стиль наших заголовков – нам придется проделывать все заново!!!! Это очень неудобно и неправильно!!! Для этого как раз разделяют между собой HTML и CSS. То есть нам достаточно повесить класс на заголовки и разом присвоить им стили!!!

!!! Чтобы расставить курсор сразу в нескольких местах, где нам необходимо – достаточно зажать **Ctrl** и щелкнуть в нужных местах – это называется **Мультикурсор**!!!

В имя класса так же необходимо вкладывать какой-то смысл, чтобы потом понимать предназначение и смысл каждого класса!!!

Например:

<h2 class=**"sectio-"**>**Обо мне**</h2>

Чтобы воспользоваться технологией CSS используя какой-то стиль есть атрибут style=**""** – он позволяет определенному html тегу придать стилистическую окраску!!!

Например:

<h2 style=**"color: blue;"**>**Я в соц.сетях**</h2>

Но не стоит злоупотреблять этим методом так как это нарушает разделение технологий!!! И чтобы поменять внешний вид элемента – придется долго искать его в коде HTML!!!

Тем не менее этот атрибут часто применяется в подобных случаях!!!

О них подробнее узнаем изучив CSS и JavaScript!!!

Чтобы одним махом в HTML сделать все заголовки определенного цвета – нужно подключить файл стилей!!!

То есть в теге верхнего уровня – <head></head> нужно разместить тег <link> для подключения сторонних файлов!!!

Атрибут rel=**""** в теге <link> служит для определения отношений между текущим документом и документом на который идет ссылка!!!

В данном случае ссылка ведет на таблицу стилей!!!!

В атрибуте href=**""** указываем путь к файлу!!!

Например:

<link rel=**"stylesheet"** type=**"text/css"** href=**"style.css"**>

То есть нужно создать этот файл

**style.css**

То есть с помощью атрибута class=**""** происходит то самое разделение технологий HTML и CSS. И очень быстро мы можем придавать и менять стиль нашей странички!!!

Еще один атрибут data=**""** используется для передачи данных с html в JavaScript!!!

Он записывается в виде

data-[значение\_min или max]= **"количество\_значений\_данных"**

data-max=**"100"**

То есть смысл атрибута data- передать какие-то значения в JavaScript.

Итог: атрибутов существует великое множество – они могут быть специфическими и универсальными!!! Они записываются в виде:

Имя\_атрибута = **"значение\_атрибута"**

**[HTML для начинающих – #9 – Формы. Введение](#_top)**

**Форма** – это то, куда мы вводим данные для отправки владельцу.

Формы могут использоваться для авторизации на сайте, введения персональной информации, имени, адреса, номера кредитной карты, задания параметров фильтрации содержимого, поиска по сайту, загрузки файлов и еще каких-либо действий!

Для создания форм используется парный тег <form></form> с возможными двумя атрибутами:

1. action = **"адрес\_файла\_формы"** – указывается файл на стороне сервера, который будет обрабатывать данные форму!!!

2. method = **"post** либо **get"**:

С помощью метода **"post"** данные передаются скрытым образом!!!

С помощью метода **"get"** данные передаются в адресной строке!!!

Например

<form action=**""** method=**"post"**>

Для создания поля ввода используется тег <input>

<input type=**""** name=**""** id=**""** value=**""**>

Который имеет три важных атрибута:

1. type = **"text или пароль"** – тип – текст или пароль.

2. name = **"имя\_поля"** – с помощью данного имени потом на сервере можно будет обращаться к данному полю.

3. id = **"имя\_id"** – с помощью этого атрибута потом на JavaScript можно будет обращаться к данному полю.

4. value=**""** – через данный атрибут передаются значения, которые позже будут отработаны на сервере!!!

В HTML5 появилось много различных типов ввода данных таких как:

1. Телефон

2. Дата

3. url

4. email

5. номер

6. цвет

7. Время UTC.

Работа формы происходит следующим образом – при нажатии на кнопку submit данные считываются с полей и отправляются на сервер файл который указан в атрибуте action=**""**

Далее на сервере по именам полей происходит считывания значений введенных в форму и их обработка.

!!!Через значение value=**"Введите имя"** – не стоит указывать подсказку ввода!!!!!

Для этого в HTML5 есть атрибут placeholder=**""**.

Подсказки отображаются серым цветом и при вводе данных печать начинается с первой буквы то есть подсказка убирается!!! А не так как в атрибуте value=**""**!!!!

Кнопка сброса:

<input type=**"reset"**>

Чтобы изменить стандартное значение **reset** на свое нужно в атрибуте value=**""** указать свое значение!!!

Например:

<input type=**"reset"** value=**"Сброс"**>

То есть при нажатии на данную кнопку значения полей очищаются!!!

В итоге HTML код формы имеет вид:

<form action=**"auth.php"** method=**"post"**>

<input type=**"text"** name=**"name"** id=**"name"** placeholder=**"Введите имя"**><!--Ввод именни пользователя-->

<input type=**"password"** name=**"pswrd"** id=**"pswrd"** placeholder=**"Введите пароль"**><!--Ввод пароля-->

<input type=**"submit"** name=**"submit"** id=**"submit"** value=**"Войти"**><!--Кнопка подтверждения отправки данных на сервер-->

<input type=**"reset"** value=**"Сброс"**><!--Кнопка сброса-->

</form>

**[HTML для начинающих – #10 – Формы. Часть 2](#_top)**

**Ctrl + D** – продублировать!!!

1. **Тип ввода** type=**"radio"** – это выбор одного из нескольких значений!!!

Например:

<input type=**"radio"** name=**"gender"** value=**"gender"**> **М**

<input type=**"radio"** name=**"gender"** value=**"fegender"**> **Ж**

Ваш пол: **М** или **Ж**!!!

Специальный тег <label></label> позволяет делать подписи к полям ввода <input> и любым элементам формы!!!

Существует два способа использования тега <label></label>:

1. Когда вводимое поле находится между открывающим и закрывающим тегом:

<label> **Имя клиента**

<input type=**"text"** name=**"client\_name"** placeholder=**"Ведите имя"**>

</label>

Удобство данного способа заключаются в том, что нажав на наш лейбл **Имя клиента** фокус автоматически перемещается в то поле которое предназначается для заполнения!!!

2. Способ не оборачивать в <label> тег <input>, а сделать ассоциацию через id (for=**"client\_name"**):

<label for=**"client\_name"**>**Имя клиента**</label>

которое должно совпадать с атрибутом id для поля (id=**"client\_name"**):

<input type=**"text"** id=**"client\_name"** name=**"client\_name"** placeholder=**"Ведите имя"**>

Для того чтобы сделать более удобным восприятие стоит разбить по строкам смысловые зоны – для этого нужно обернуть каждый <label> <input> в блок <div>

**Ctrl + Alt + Enter**!!!

Плагин Emmet существенно ускоряет процесс набора кода!!!

Его обязательно нужно добавить в Sublime text!!! и изучить все его фишки!!!

2. **Элемент выбора нескольких значений** – Checkbox – это разновидность тега <input> как и радио type=**"radio"**!!!

Атрибут checked ставится в том чекбоксе который должен быть по умолчанию!!!

3. **Элемент** select – это выпадающий список, он имеет вид:

<span>**Тип сайта**</span>

<select name=**"type"**>

<option value=**"eshop"**>**Интернет магазин**</option>

<option value=**"landing"**>**Лэндинг**</option>

<option value=**"card"**>**Сайт-визитка**</option>

</select>

Атрибут множественного выбора multiple позволяет из всего выпадающего списка <select></select> выбрать несколько значений!!!

Только для этого нужно зажать кнопку **Ctrl + кликнуть мышью на желаемых пунктах**!!!

**Ctrl + Alt + F** – авто форматирование!

4. **Элемент поле** – для выбора файла, HTML код которого имеет вид:

Первый способ:

<input type=**"submit"** name=**"Отправить"**>

Второй способ:

<button name=**"submit"**>**Отправить**</button>

5. **Тег** <input type=**"hidden"** value=**""**>– этот параметр не отображается на сайте но через значения value=**""** мы можем передать на сервер скрытые параметры!!!

Например – сколько пользователь находился времени на нашем сайте?!!!

<input type=**"hidden"** name=**"time"** value=**"time\_count"**>

И при нажатии кнопки **"submit"** скрытые данные отправятся на сервер!!!

6. **Тег текстовое поле**:

<textarea name=**"comment"** id=**""** cols=**"30"** rows=**"10"** maxlength=**"100"**></textarea>

где cols=**"30"** – количество столбцов!

rows=**"10"** – количество строк в поле.

maxlength=**"100"** – максимальное количество введенных символов!!!

**[HTML для начинающих – #11 – Семантические особенности HTML5](#_top)**

1. Помимо семантических особенностей HTML5 принес множество новшеств для упрощений создания графических и мультимедийных объектов без применения сторонних плагинов. Это например – объекты видео и аудио и возможности рисования с помощью SVG и Canvas.

Чтобы добавить новую заготовку используя плагин Emmet в Sublime text нужно нажать ! и Tab -> и получим готовую HTML5 заготовку:

<!DOCTYPE html>

<html lang=**"en"**>

<head>

<meta charset=**"UTF-8"**>

<title>**Document**</title>

</head>

<body>

**...**

</body>

</html>

Раньше в HTML5 был строгий режим записи DOCTYPE:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">

Который говорил, что данный документ не содержит элементов, помеченных как устаревшие!!! Или неодобряемые!!!

Или был переходный режим Transitional:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

Но в HTML5 это сильно упростили, чтобы не запоминать это все и не искать откуда его скопировать:

<!DOCTYPE html>

2. Поддержка новых тегов старыми браузерами, в частности, IE8. Для этого есть специальная конструкция называемая условными комментариями, которая говорит, что если ваша версия браузера меньше IE9 то надо использовать специальный скрипт **html5shiv** и представляет из себя **полифил**, который создает элементы с названиями новых тегов для того чтобы старые браузеры не воспринимали их как неизвестные, а должным образом их обрабатывали!!!

<--! [if It IE 9]>

<script src=**"//html5shiv.google.com/svn/trunk/html5.js><script>**

**<![endif]-->**

Не смотря на то, что в HTML5 добавлено достаточно много новых тегов – на практике редко встретить всех их обрабатываемые в одном документе!!!

Применение всех тегов сразу и желание использовать отдельный тег на каждый случай жизни наоборот ухудшает читаемость кода и создает кашу из слишком большого числа разных тегов!!!

Чаще всего встретятся следующие теги, которые отвечают за основу разметки страницы:

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**|** <header> **|**

**|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|**

**|** <nav> **|** <section> **|** <aside> **|**

**| |** <header> **| |**

**| |** <arcticle> **| |**

**| |** <footer> **| |**

**|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|**

**|** <footer> **|**

**|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|**

Прям сейчас их уже стоит активно применять!!!

Первое что стоить добавить – это шапка сайта или header.

Для этого используется специальный тег <header></header>

Специальный тег <address></address> в блоке <header></header> – специально сделан для поисковиков и они лучше понимают что в данной части находиться контактная информация с владельцем этого сайта!!!

Например:

<header>

<h1 align=**"center"**>**Мой сайт портфолию**</h1>

<address>

<div>**Город: Москва**</div>

<div>**Телефон: 8-123-456-89**</div>

</address>

</header>

Логические блоки сайта можно сворачивать, чтобы лучше смотреть из чего состоит сайт!!!

Он состоит из:

1. Формы авторизации.

2. Я в соц.сетях.

3. Информация обо мне

4. Биография и т.д.

**Логические блоки** – представляют как раз основную информация сайта и подобно тегам <header></header> и <footer></footer> был создан тег <main></main>, который говорит о том, что здесь будет идти основная часть содержимого сайта!!!

Встречаться он может только одни раз и не может быть обернут в какие-то другие теги типа <section></section> или <aside></aside>.

Все вспомогательные блоки, например – Я в соц.сетях, авторизация или дерево хэштегов и прочее отделяется в специальный блок <aside></aside>!!!! Который говорит, что здесь идет второстепенная информация – вспомогательная информация!!! Визуально – в виде каких-то колоночек отображается!!!

Правильной практикой является разделение на независимые блоки логических структур, например, форма авторизации должна быть обернута в отдельный блок <div></div>. И точно также другие логические блоки нужно отделять друг от друга!!!

Тег <nav></nav> специально предназначен для меню!!!

Например:

<nav>

<li><a href=**"#"**>**Главная**</a></li>

<li><a href=**"#"**>**Портфолию**</a></li>

<li><a href=**"#"**>**Блог**</a></li>

</nav>

Атрибут href=**"#"** означает что ссылки еще не созданы и **"#"** является некой заглушкой!!!

Если свернуть все базовые элементы – это <section>, <aside>, <nav> в одну прямую, чтобы увидеть весь сайт в масштабе!!!

В теге <footer></footer> пишется информация о правах на владение сайта!!!

Чтобы информацию на нем не копировали.

Также в футер можно поместить контактную информацию обернув ее в тег <adress></adress>.

Например:

<footer>

<div>**Это мой сайт. Пожалуйста не копируйте его.**</div>

<address>

<div>**Город: Москва**</div>

<div>**Телефон: 8-123-456-89**</div>

</address>

</footer>

То есть сайт состоит из:

1. Шапки

2. Подвала

3. Основной части, в которой содержаться:

1) Вспомогательной информации

2) Навигации по сайту.

3) Три блока секции <section></section>

Чтобы разделить левую колонку от правой и сделать хорошую структурированность – нужно обернуть вспомогательную информацию и навигацию в блок для левой колонки!!!

И всю остальную текстовую информацию – в блок для правой колонки!!!

Для таких случаев не предусмотрено каких-то специальных тегов и вообще существует негласное правило, что:

**"Если не знаешь в какой тег HTML5 обернуть то оборачивай просто в <div></div>."**

Здесь не будет никакой грубой ошибки – это будет лучше, чем применить не тот тег!!!

И в дальнейшем к этим блокам будут применены свойства CSS визуально сайт обретет должный вид!!!

Тег для статей <aricle></aricle> – очень часто применяется в блогах, когда отделяется одна статья от другой!!!

Тег <figure></figure> – с помощью которого можно группировать изображения и подписи к ним!!!

Подписи к изображениям делаются с помощью тега <figcaption></figcaption>.

Например:

<figure>

<img src=**"http://lorempixel.com/200/200/sports"** alt=**"Мой аватар"** title=**"Петр Омельницкий"** width=**"200"** height=**"200"** border=**"2"**>

<figcaption>

Мое фото

</figcaption>

</figure>

Теперь когда html разметка представляет из себя хорошо структурированный семантичный html документ – дольше идет визуальное оформление странички с помощью CSS.

Большую часть новшеств html5 принес не в разметку, а в язык веб разработки JavaScript.