

# Верстка web-страниц

## Введение

Титова Ольга Ивановна  
Минск, 2017



# Содержание

1. Основные понятия
2. Инструментарий
3. История
4. Сведения о HTML и CSS
5. HTML документ и его структура



# Основные понятия

**Веб-страница** - документ или информационный ресурс Всемирной паутины, доступ к которому осуществляется с помощью веб-браузера.

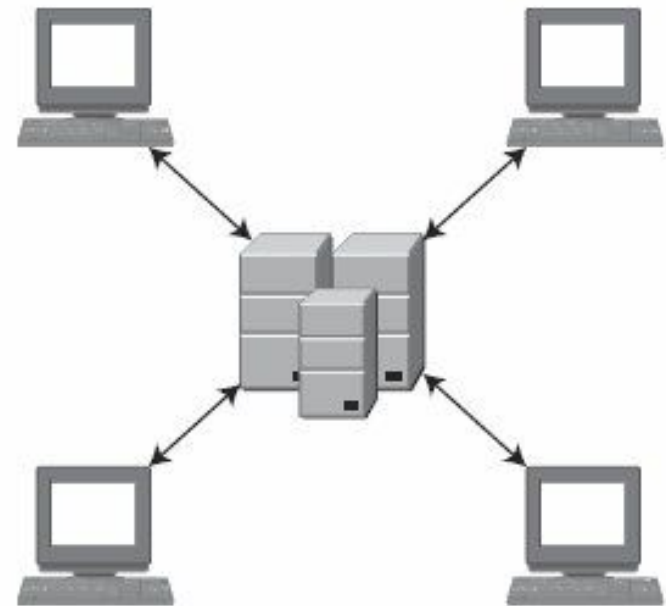
Бывают двух типов: статические и динамические.



# Основные понятия

**Сервер** - программное обеспечение, принимающее запросы от клиентов.

**Клиент** - в нашем случае браузер.



# Основные понятия

**HTTP** - (англ. HyperText Transfer Protocol — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных (изначально — в виде гипертекстовых документов). Основой HTTP является технология «клиент-сервер».



# Основные понятия

**Сайт** - (от англ. website: web — «паутина, сеть» и site — «место», буквально «место, сегмент, часть в сети») — совокупность электронных документов (файлов) частного лица или организации в Сети.



# Структура сайта

## **Логическая –**

информационное структурирование (что, где, в каком объеме будет размещено, каким образом будет выглядеть система навигации, доступ к каким объектам и каким образом будет организован и т.п.)

## **Физическая –**

техническая реализация спроектированной информационной структуры:

- формирование файловой системы для сайта;
- определение стартовой страницы;
- подключаемые СУБД и БД;
- и т.п.



# Основные понятия

**URL** - (англ. URL — Uniform Resource Locator) — единообразный локатор (определитель местонахождения) ресурса. Ранее назывался Universal Resource Locator — универсальный указатель ресурса. URL — это стандартизированный способ записи адреса ресурса в сети Интернет.

Или просто **ссылка** 😊





# Основные понятия

**HTML** - (HyperText Markup Language) — язык гипертекстовой разметки.

**Документ HTML** представляет в своей основе текстовый документ, в котором имеются специальные символы – теги, определяющие структуру документа и позволяющие осуществлять связь между несколькими подобными документами.

Именно из-за возможности размещать в документе ссылки на другие документы и с их помощью осуществлять переходы данный язык называется **гипертекстовым**.



# Основные понятия

**HTML** – структурирует документ, упорядочивая информацию в структурные единицы – заголовки, абзацы, списки и другие элементы.

**CSS** – взаимодействуя с браузером, придает документу определенный стилевой формат – цвет, размер, отступы и другие оформительские свойства.

**Пример,**

**Объект** – фраза из нескольких слов

**HTML** – тип объекта (абзац или заголовок), последовательность при размещении в общем контенте.

**CSS** – тип шрифта, размер шрифта, цвет начертания, наличие обрамления текста рамкой, отступ от рядом расположенных объектов.



# Основные понятия

## Преимущества CSS:

- больше возможностей форматирования;
- применение одного свойства к произвольному количеству однотипных объектов;
- описание свойств с использованием стилей значительно уменьшает суммарный объем разрабатываемого ресурса;
- упрощается процесс обновления ресурса при внесении стилевых изменений.



# Инструментарий

Для начальной разработки веб-страниц или даже небольшого *сайта* (набор страниц, связанных между собой ссылками и единым оформлением), нам понадобятся **следующие программы:**

Текстовый редактор.

Браузер(ы) для просмотра результатов.

Валидатор — программа для проверки синтаксиса HTML и выявления ошибок в коде (для CSS также).

Графический редактор.

Справочная литература.

Это необходимый **минимум**



# История HTML

**SGML** – предок HTML. В свое время был разработан для совместного использования машинно-читаемого текста в больших правительственных программах (например, аэрокосмических).

**HTML** – разработан в конце 1980-х годов. Создавался для обмена научной и технической информацией между университетами, давая возможность работать с ним без серьезной профессиональной подготовки.



# История HTML

**HTML 2.0** – одобрен как стандарт в 1995

От версии к версии, до текущего **HTML 5** добавлялись дополнительные возможности представления информации на веб-странице.

**XHTML 1.0** – оформлен как стандарт в 2000 году. Смесь **XML** + **HTML 4**, сделанная ради стандартизации разработки веб-страниц. Идею признали несостоятельной, и почти приняв **XHTML 2.0**, разговоры о продолжении стандарта прекратились. Вместо этого силы были брошены на разработку **HTML 5**.



# Развитие HTML

Основные причины развития HTML:

1. Развитие мультимедиа-возможностей ПК
2. Появление карманных компьютеров и смартфонов, планшетов
3. Необходимость стандартизации
4. Упрощение разработки

За всем этим следит **Консорциум Всемирной Паутины (World Wide Web Consortium, W3C)**, который был создан в 1994 году.



# Стандарт сегодня

## **Стандарт веб-разработки сегодня:**

HTML 4.1 (HTML 5 уже практикуется), CSS 2.1 (CSS 3.0)





# Теги HTML

Главным элементом разметки является **тег**.

Он не отображается непосредственно в браузере, но влияет на то, каким образом отображаются остальные элементы – текст, ссылки, изображения и другой контент.

**<тег атрибут1="значение" атрибут2="значение">**

**<тег атрибут1="значение" атрибут2="значение">...</тег>**



# Теги HTML

`<тег></тег>` - парный тег (контейнер)

`<тег />` - непарный тег

## Примеры:

`<title>Заголовок страницы</title>`

`<strong>Внутри этого тега все символы выделены  
жирным</strong>`


`` - откуда и какую картинку необходимо  
взять браузеру для отображения на странице



# Теги HTML

## Парный тег (контейнер)

Парные теги, называемые по-другому контейнеры, состоят из двух частей — открывающий и закрывающий тег. Открывающий тег обозначается как и одиночный — **<тег>**, а в закрывающем используется слэш — **</тег>**. Допускается вкладывать в контейнер другие теги, однако следует соблюдать их порядок.



The diagram illustrates the nesting of HTML tags. It shows the code: `<div><p>Lorem <b>ipsum dolor</b> sit amet...</p></div>`. Three brackets are drawn above the code to show the scope of each tag: a blue bracket for the outermost `<div>` tag, a green bracket for the `<p>` tag, and a red bracket for the `<b>` tag. The brackets are nested, with the red one inside the green one, and the green one inside the blue one, demonstrating how inner tags are contained within outer ones.

```
<div><p>Lorem <b>ipsum dolor</b> sit amet...</p></div>
```



# Теги HTML

Если связать открывающий и закрывающий тег между собой скобкой, как показано на рис., то несколько скобок обозначающих разные контейнеры, не должны пересекаться между собой. Любое пересечение условных скобок говорит о том, что правильная последовательность тегов нарушена.



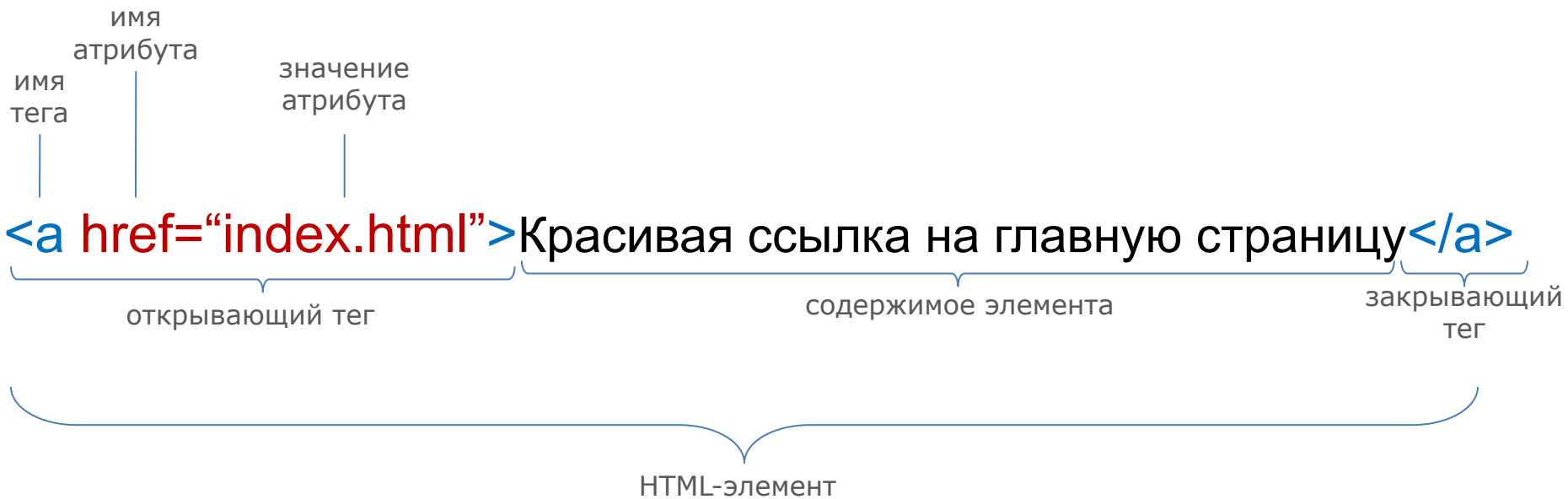
# Теги HTML

## Теги:

1. Обрамляются `<>`.
2. В теге могут находиться атрибуты (дополнительные параметры для элементов). Атрибуты могут быть практически у любого тега.
3. Содержимое тега пишется между открывающим и закрывающим тегами. Если тег непарный, то содержимое пишется в атрибуте.
4. Теги могут следовать линейно, а могут располагаться вложено (на подобие вложенных скобок).



# Теги HTML



# Теги HTML

Для тегов любого типа действуют определенные правила их использования.

Причем, некоторые правила обязательны для выполнения, а другие являются рекомендациями, т.е. их можно выполнять, а можно и нет.



# Теги HTML

## **Атрибуты тегов и кавычки**

Согласно спецификации HTML все значения атрибутов тегов следует указывать в двойных ("пример") или одинарных кавычках ('пример').

Отсутствие кавычек не приведет к ошибкам, браузеры во многих случаях достаточно корректно обрабатывают код и без кавычек, за исключением текста, содержащего пробелы.





# Теги HTML

**Теги можно писать как прописными, так и строчными символами**

Любые теги, а также их атрибуты нечувствительны к регистру, поэтому вы вольны выбирать сами, как писать — **<BR>**, **<Br>** или **<br>**. В любом случае рекомендуется придерживаться выбранной формы записи на протяжении всех страниц сайта.



## Переносы строк

Внутри тега между его атрибутами допустимо ставить перенос строк. Однако, при этом следует учитывать «читабельность» и визуальное восприятие кода – лучше, когда код воспринимается в той логической структуре, в которой он создавался.



## Неизвестные теги и атрибуты

Если какой-либо тег или его атрибут был написан неверно, то браузер проигнорирует подобный тег и будет отображать текст так, словно тега и не было.

Опять же, следует избегать неизвестных тегов, поскольку код HTML не пройдет валидацию.



# Теги HTML

## Порядок тегов

Существует определенная иерархия вложенности тегов. Например, тег **<title>** должен находиться внутри контейнера **<head>** и нигде иначе. Чтобы не возникло ошибки, следите за тем, чтобы теги располагались в коде правильно.

Если теги между собой равноценны в иерархии связи, то их последовательность не имеет значения. Так, можно поменять местами теги **<title>** и **<meta>**, на конечном результате это никак не скажется.



# Структура документа

`<!DOCTYPE HTML>` - обозначение типа документа для браузера // в данном случае определение типа документа для HTML5 самое простое

`<html>` - начало документа

`<head>` - служебная часть документа

`<title>Заголовок страницы</title>`

`</head>`

`<body>` - содержательная часть документа

`</body>`

`</html>` - конец документа



# Структура документа

```
<!DOCTYPE html PUBLIC
```

```
"-//W3C//DTD HTML 4.01 Transitional//EN"
```

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

- таким образом выглядит обозначение типа документа для стандартной страницы на HTML4.01



# Структура документа

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
  <title>Заголовок страницы</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```



# Служебная информация

1. Тег `<title>` с заголовком страницы
2. Теги `<meta />` - в основном используются для браузеров и поисковых систем
3. Теги `<link />` - подключение CSS
4. Теги `<script>` - подключение JS





# Работа с документом

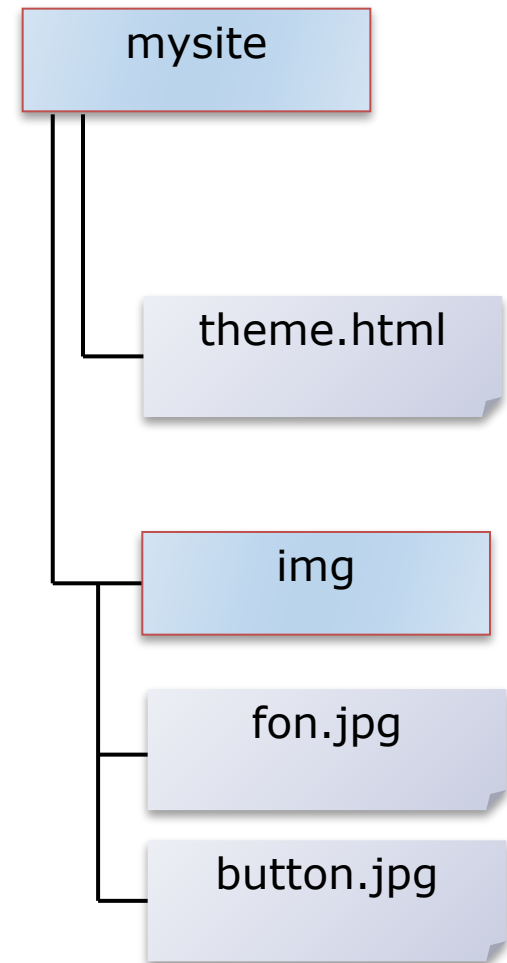
1. Создавать и редактировать файлы можно в любом текстовом редакторе (к примеру, NotePad++)
2. Шаблон имени файла: \*.html – ваш компьютер поймет, что этот файл нужно открывать в браузере
3. В имени файла только латинские буквы, и цифры, подчеркивания(\_) и дефисы (-).
4. Русские буквы, пробелы и спец.символы (%\$#) не рекомендуется использовать.
5. Все файлы по сверстанному сайту размещайте в отдельной системе вложенных папок.



# Создание документа

1. Создайте каталог **mysite**
2. Откройте Notepad
3. Сохраните пустой файл под именем **theme.html** в каталоге **mysite**
4. Откройте полученный файл и приготовьтесь верстать 😊
5. Для используемых изображений (к примеру, fon.jpg, button.jpg) создайте вложенную папку, к примеру, img.

**! Не допускайте хаоса в файловой структуре**



# Редактирование

Напишите в документе  
theme.html

```
<!doctype html>
```

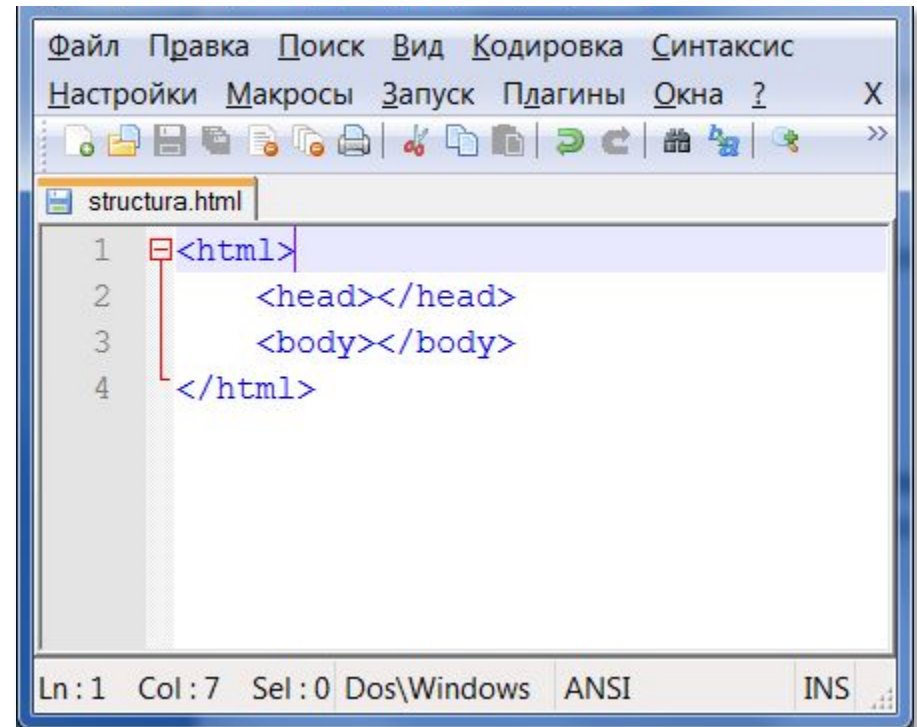
```
<html>
```

```
    <head></head>
```

```
    <body></body>
```

```
</html>
```

Если сейчас открыть  
файл в браузере: он  
будет пустой.



```
Файл  Правка  Поиск  Вид  Кодировка  Синтаксис
Настройки  Макросы  Запуск  Плагины  Окна  ?  X
[Icons]
structura.html
1  <html>
2      <head></head>
3      <body></body>
4  </html>
Ln: 1  Col: 7  Sel: 0  Dos\Windows  ANSI  INS
```



# Редактирование

1. Добавьте между тегами <head> тег

<title>:

<title>Моя первая HTML

страница</title>

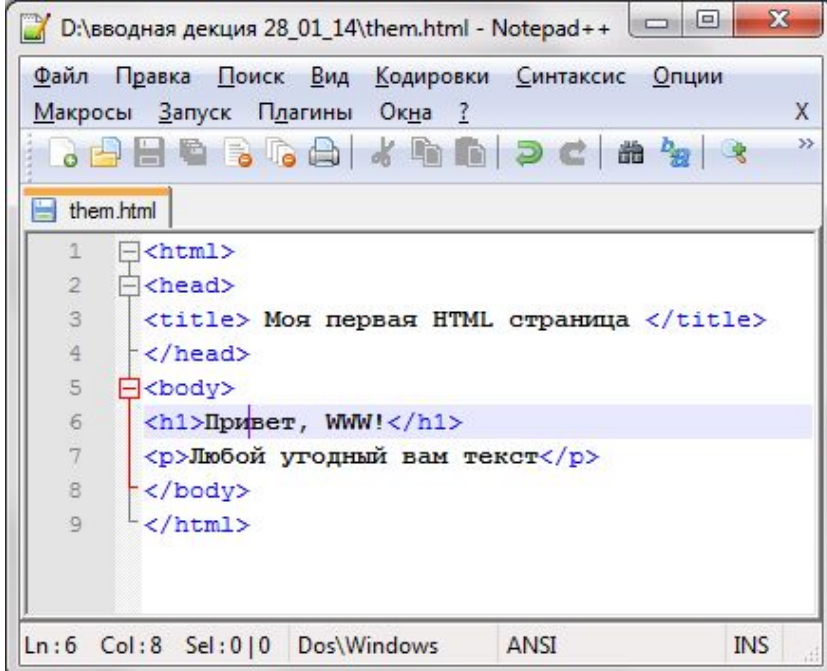
2. Добавьте между тегами <body>

теги <p> и <h1>:

<h1>Привет, WWW!</h1>

<p>Любой удобный вам текст</p>

3. Обновите страницу в браузере.



The screenshot shows a Notepad++ window titled "D:\вводная лекция 28\_01\_14\them.html - Notepad++". The menu bar includes "Файл", "Правка", "Поиск", "Вид", "Кодировки", "Синтаксис", and "Опции". The toolbar contains various icons for file operations and editing. The main text area shows the following HTML code:

```
1 <html>
2 <head>
3 <title> Моя первая HTML страница </title>
4 </head>
5 <body>
6 <h1>Привет, WWW!</h1>
7 <p>Любой удобный вам текст</p>
8 </body>
9 </html>
```

The status bar at the bottom indicates "Ln: 6 Col: 8 Sel: 0 | 0 Dos\Windows ANSI INS".



# Комментарии

HTML позволяет оставлять в коде страницы комментарии (например, они нужны для объяснения какой-либо части кода, чтобы в будущем не ломать себе голову, зачем вы это сделали).

В браузере комментарии не отображаются.

```
<!-- Комментарий -->
```

```
<!--
```

```
<h1>Закомментированный заголовок страницы</h1>
```

```
-->
```



# Теги HTML

Условно теги делятся на **следующие типы**:

теги верхнего уровня;

теги заголовка документа;

блочные элементы;

строчные элементы;

универсальные элементы;

списки;

таблицы;

фреймы.



# Теги HTML

Каждый тег HTML принадлежит к определенной группе тегов, например, табличные теги направлены на формирование таблиц и не могут применяться для других целей.

Следует учитывать, что один и тот же тег может одновременно принадлежать разным группам, например, теги **<ol>** и **<ul>** относятся к категории списков, но также являются и блочными элементами.



# Теги HTML

## Теги верхнего уровня

Эти теги предназначены для формирования структуры веб-страницы и определяют разделы заголовка и тела документа.

### **<html>**

Тег **<html>** является контейнером, который заключает в себе всё содержимое веб-страницы, включая теги **<head>** и **<body>**. Открывающий и закрывающий теги **<html>** в документе необязательны, но хороший стиль диктует неизменное их использование.





# Теги HTML

## **<head>**

Тег **<head>** предназначен для хранения других элементов, цель которых — помочь браузеру в работе с данными.

Также внутри контейнера **<head>** находятся метатеги, которые используются для хранения информации, предназначенной для браузеров и поисковых систем. Например, механизмы поисковых систем обращаются к метатегам для получения описания сайта, ключевых слов и других данных.



# Теги HTML

## **<body>**

Тег **<body>** предназначен для хранения содержания веб-страницы, отображаемого в окне браузера. Информацию, которую следует выводить в документе, следует располагать именно внутри контейнера **<body>**. К такой информации относятся текст, изображения, таблицы, списки и др.



# Теги HTML

## Теги заголовка документа

К этим тегам относятся элементы, которые располагаются в контейнере **<head>**. Все эти теги напрямую не отображаются в окне браузера, за исключением тега **<title>**, который определяет название веб-страницы.

### **<title>**

Используется для отображения строки текста в левом верхнем углу окна браузера, а также на вкладке. Такая строка сообщает пользователю название сайта и другую информацию, которую добавляет разработчик.



## Теги заголовка документа

### **<meta>**

Метатеги используются для хранения информации, предназначенной для браузеров и поисковых систем. Например, механизмы поисковых систем обращаются к метатегам для получения описания сайта, ключевых слов и других данных. Хотя тег **<meta>** всего один, он имеет несколько атрибутов, поэтому к нему и применяется множественное число.



# Теги HTML

## ВАЖНО

Работая с html-кодом, который придает особый вид контенту, разбивая его на логические блоки, **думайте о структуре**

У веб-страницы должна быть **логическая структура**, которую и задают теги, используемые **для разметки** содержимого



# Правила хорошего тона

## **Под капотом:**

1. Код должен быть «валидным», правильным, написанным по правилам. Проверить можно на странице W3C - <http://www.validator.w3.org>
2. Хоть браузеру и все равно, но теги лучше писать строчными буквами
3. При необходимости не жалейте комментариев

## **В браузере:**

1. Правило трех шрифтов
2. Мелькающие картинки – это плохо, тем более на фоне страницы
3. Цвет шрифта контрастен к цвету фона
4. Видео или аудио не должны включаться автоматически. Только сам пользователь включает.



# Полезные ссылки

<http://www.validator.w3.org> - проверка «валидности» верстки

<http://htmlbook.ru> - все HTML теги в одном месте, с примерами

[http://ru.wikipedia.org/wiki/Война\\_браузеров](http://ru.wikipedia.org/wiki/Война_браузеров) - немного истории

<http://ru.wikipedia.org/wiki/HTTP> - HTTP



Спасибо за внимание

