






ОСНОВЫ HTML



Елена Иванова
@liveldi90
Front-end developer
Artec3D

Содержание

-  История HTML.
-  Структура HTML документа.
-  Изучаем теги и их атрибуты.
-  Семантика в HTML и почему это важно.
-  Практика.

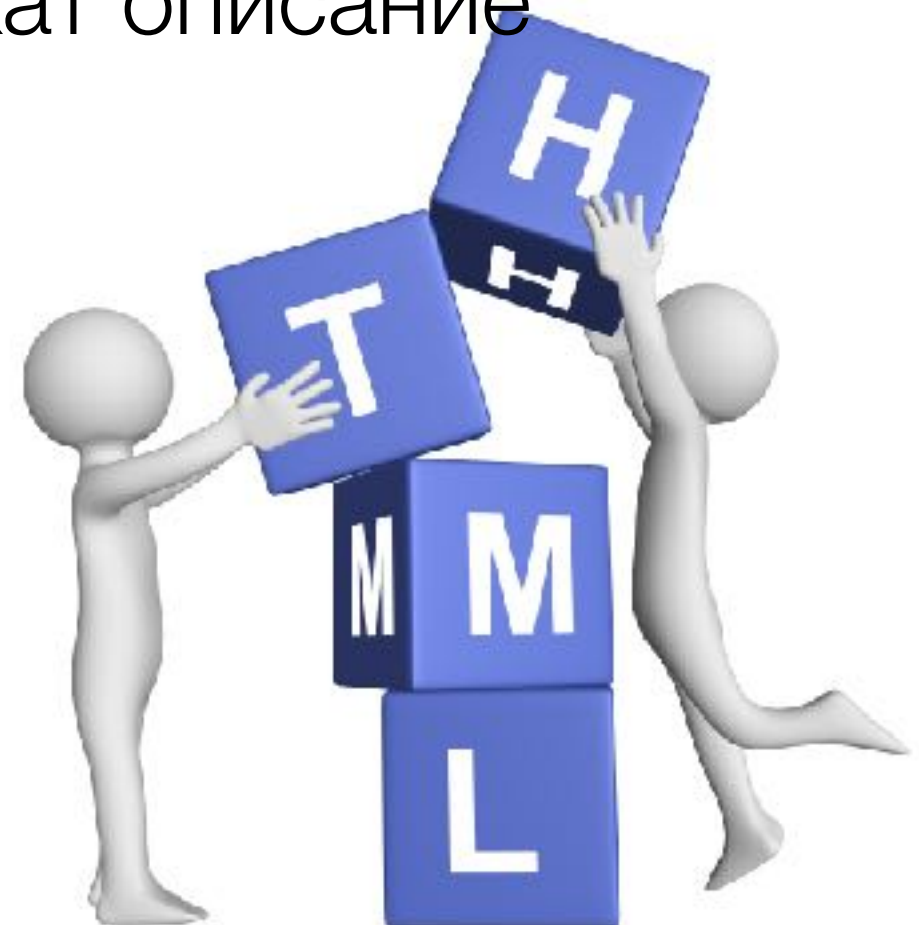
История HTML

Что такое HTML?

*HyperText Markup Language — «язык
гипертекстовой разметки»*

Стандартизированный язык разметки документов во Всемирной паутине.

Большинство веб-страниц содержат описание разметки на языке HTML



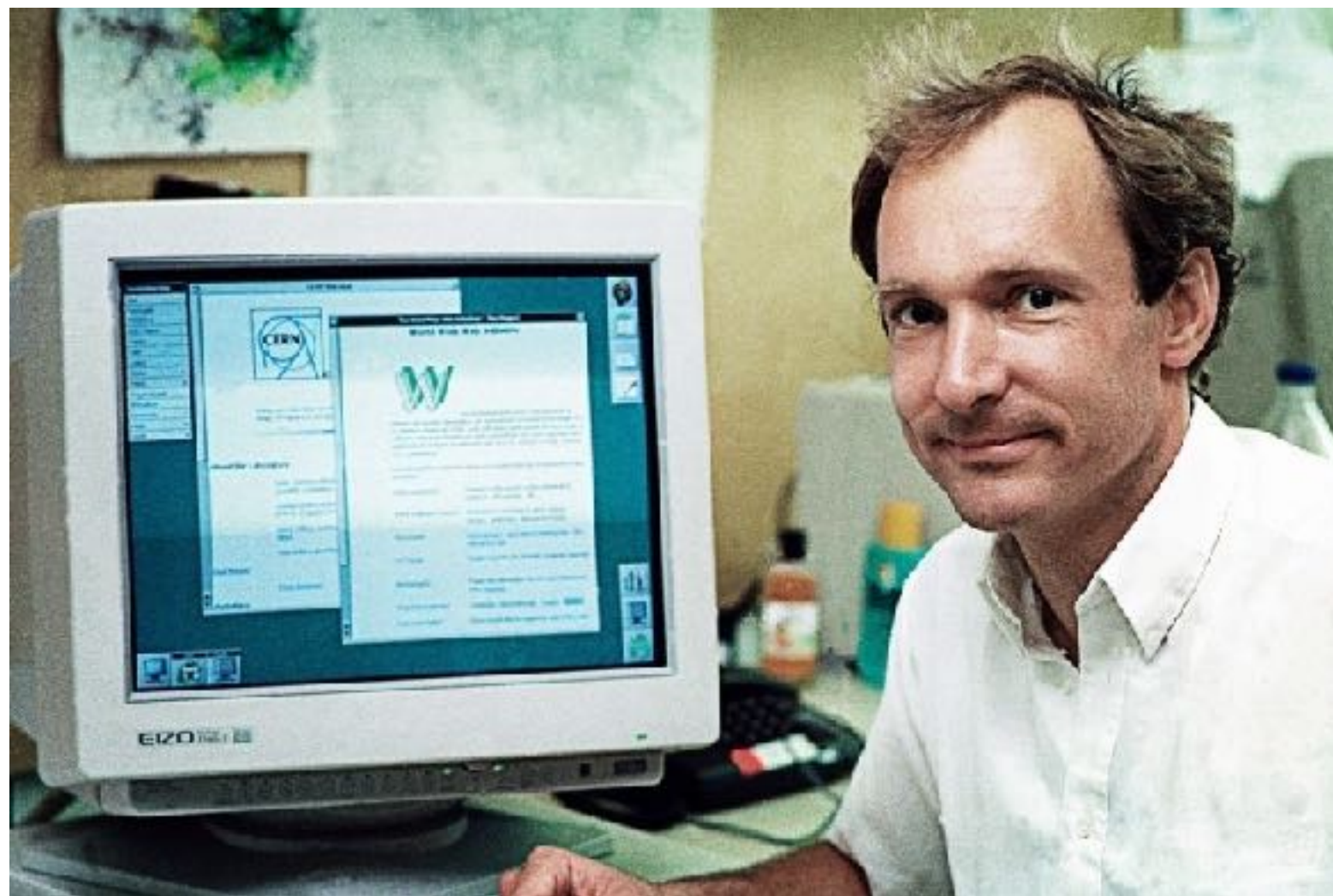
Пример HTML в devTools

(lessons4.liveldi.ru)

```
<!DOCTYPE html>
<html lang="ru">
  ▼<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-
scale=1, user-scalable=no">
    <link href="https://fonts.googleapis.com/css?family=Open+Sans:
300,400,600,700&subset=cyrillic" rel="stylesheet">
    <link href="/styles/build.css?647" rel="stylesheet" type="text/css">
    <title>MCS – кодировка для девушек</title>
  </head>
  ▼<body cz-shortcut-listen="true">
    ►<div class="container">...</div>
    ►<div class="copyright">...</div>
    <script src="/app.js?865"></script>
  </body>
...</html> == $0
```

Тимоти Джон Бернерс-Ли

— британский учёный, изобретатель HTML, создатель Всемирной паутины (совместно с Робертом Кайо) и действующий глава Консорциума Всемирной паутины (W3C).



История HTML

1986

— Появление первых версий HTML.
Выделение структурных элементов:
Заголовки, списки, абзацы

1991

— HTML стал использоваться именно
для передачи гипертекста по просторам
всемирной паутины

1994

— Создание организации W3C
(World Wide Web Consortium)



История HTML

1995

— HTML 2.0. Формы.

1997

— HTML 3.2. Таблицы, изображения, фреймы. Также появляется CSS.

1997

— HTML 4.0. Ушли устаревшие теги для оформления. Объектная модель документа. Возможность работы HTML с JS.

1999

— HTML 4.1

2014

— HTML 5. Много новых тегов, работа с видео, рисование на холсте, динамические элементы.

История HTML

1995

— HTML 2.0. Формы.

1997

— HTML 3.2. Таблицы, изображения, фреймы. Также появляется CSS.

1997

— HTML 4.0. Ушли устаревшие теги для оформления. Объектная модель документа. Возможность работы HTML с JS.

1999

— HTML 4.1

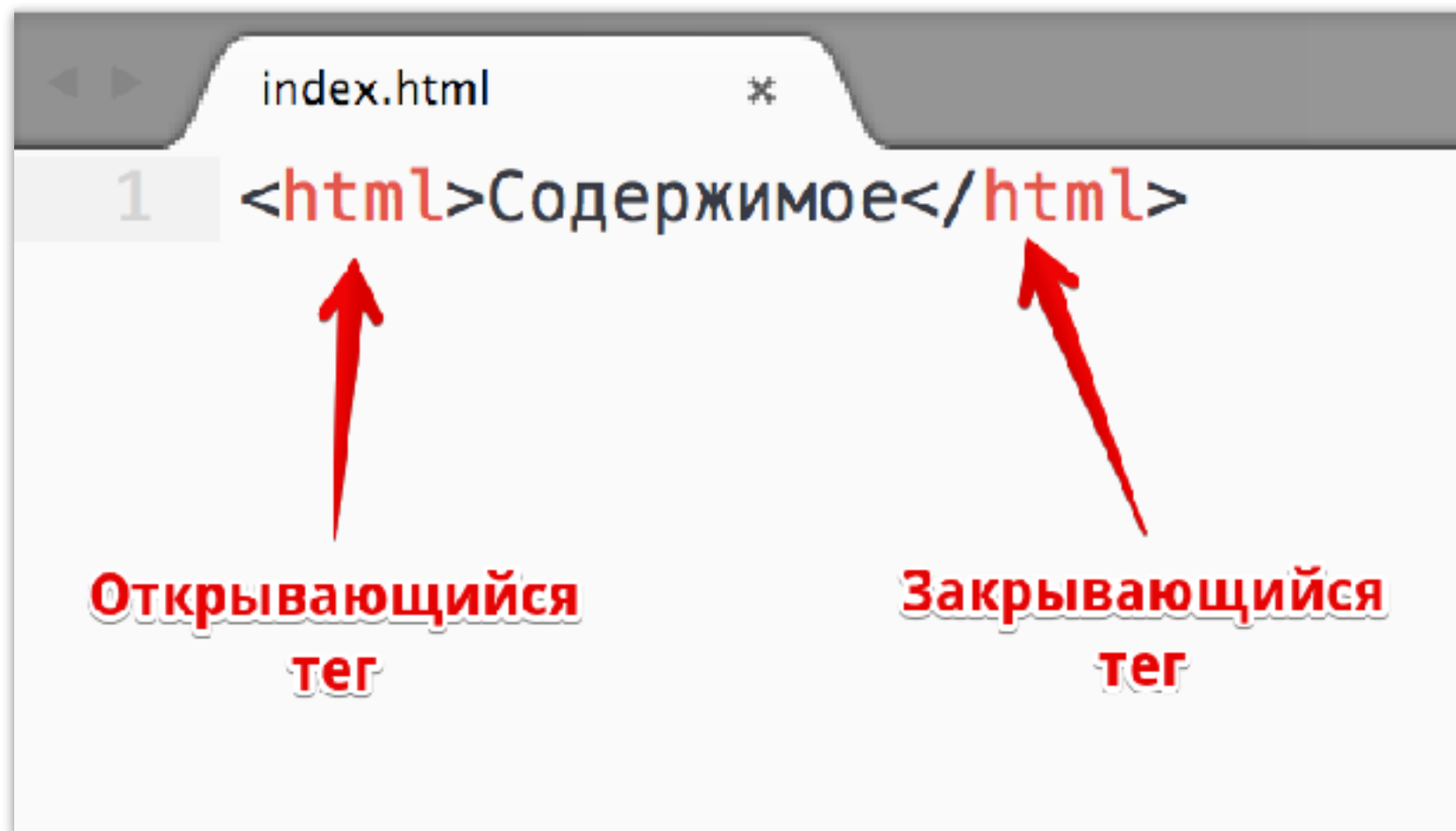
2014

— HTML 5. Много новых тегов, работа с видео, рисование на холсте, динамические элементы.

Структура HTML документа

Тег

— это метки, которые вы используете для указания браузеру, как он должен показывать ваши html-странички!



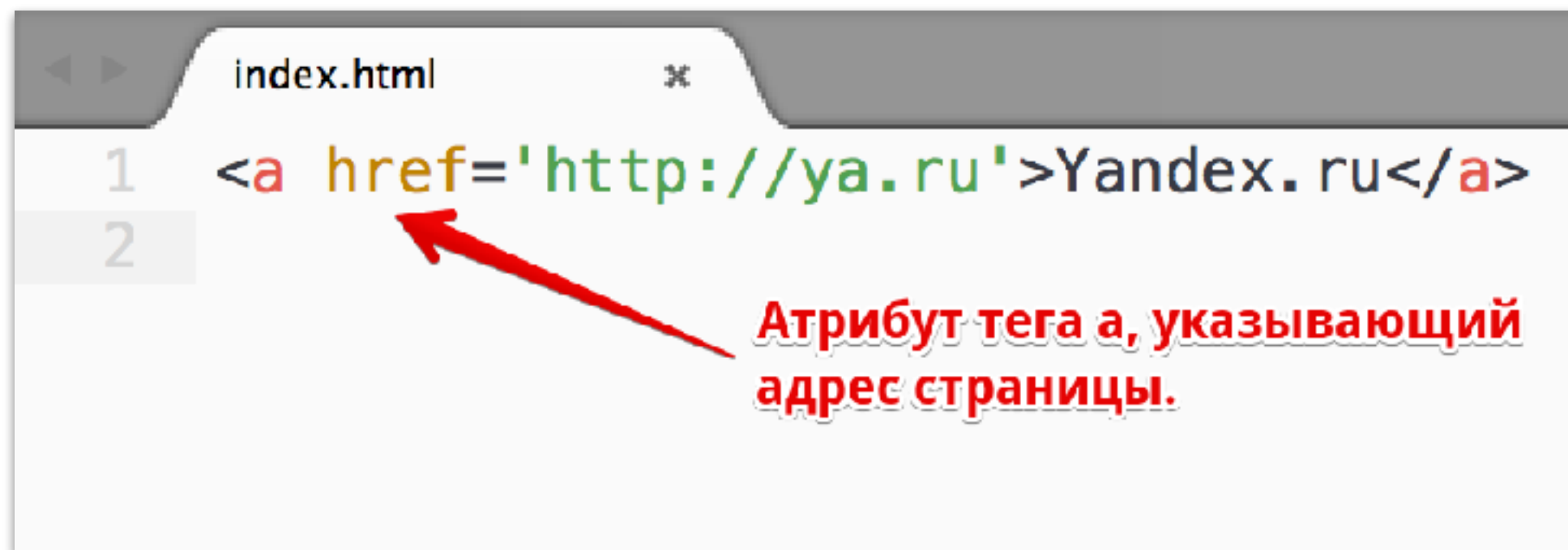
Тег (элемент)

- Обычно имеются два тэга - открывающий: **<html>** и закрывающий: **</html>**
- Бывают исключения для определенных тегов, которые сразу закрываются и не имеют содержимого, например, перенос строки выглядит так: **
**.

HTML — это тэги, и ничего кроме тэгов. Для изучения HTML нужно изучить различные тэги.

Атрибут тега

— В некоторых тэгах вы можете вводить дополнительную информацию. Такая дополнительная информация называется **"атрибут"**.



Существуют атрибуты которые присущи только определенным тегам или группам тегов, и те, которые можно добавлять почти в каждый тег.

Элемент <!DOCTYPE>

Элемент <!DOCTYPE> предназначен для указания типа текущего документа.

Это необходимо, чтобы браузер понимал, как следует интерпретировать текущую веб-страницу, поскольку HTML существует в нескольких версиях

```
<!DOCTYPE html>
```

Тег <html>

Тег <html> является контейнером, который заключает в себе все содержимое веб-страницы, включая теги <head> и <body>

```
<!DOCTYPE html>  
<html>  
  
</html>
```


Тег <head>

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

Содержит различные мета теги, стили, скрипты и др.

Напрямую не отображается на веб-странице.

```
<!DOCTYPE html>  
<html>  
  <head>  
  
  </head>  
</html>
```

Тег <meta>

<meta> определяет метатеги, которые используются для хранения информации предназначенной для браузеров и поисковых систем.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
</html>
```

Кодировка

— Совокупность комбинаций кодов (нулей и единиц) и СИМВОЛОВ, им соответствующим.

Примеры: utf-8, ASCII, Windows-1251.

Установка кодировки для html страницы обязательна, иначе браузер будет ее угадывать и может показать иероглифы :(

Самая распространённая современная кодировка — ***utf-8***

```
<meta charset="utf-8">
```

Тег <title>

Определяет заголовок документа. Элемент <title> не является частью документа и не показывается напрямую на веб-странице.

Его можно увидеть как заголовок вкладки.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Моя первая html-страничка</title>
  </head>
</html>
```

Тег <body>

Элемент <body> предназначен для хранения содержания веб-страницы (контента), отображаемого в окне браузера

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Моя первая html-страничка</title>
  </head>
  <body>
    Hello, World!
  </body>
</html>
```

Структура HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Моя первая html-страничка</title>
  </head>
  <body>
    Hello, World!
  </body>
</html>
```



Изучаем теги и их
атрибуты

Комментарии

— Позволяют писать личные заметки в коде, не будут отображены в браузере.

`<!-- Комментарий -->`

`<!-- Комментарий -->`



Заголовки

<h1>Заголовок 1-го уровня</h1>
<h2>Заголовок 2-го уровня</h2>
<h3>Заголовок 3-го уровня</h3>
<h4>Заголовок 4-го уровня</h4>
<h5>Заголовок 5-го уровня</h5>
<h6>Заголовок 6-го уровня</h6>



Параграфы

`<p>`

Элемент `body` предназначен для хранения содержания веб-страницы (контента), отображаемого в окне браузера. Информацию, которую следует выводить в документе, следует располагать именно внутри контейнера `body`. К такой информации относятся текст, изображения, теги, скрипты JavaScript и т.д.

`</p>`

`<p>`

Элемент `body` предназначен для хранения содержания веб-страницы (контента), отображаемого в окне браузера. Информацию, которую следует выводить в документе, следует располагать именно внутри контейнера `body`. К такой информации относятся текст, изображения, теги, скрипты JavaScript и т.д.

`</p>`

Списки — не нумерованный

```
<ul type="disc или circle или square">  
  <li>Пункт 1</li>  
  <li>Пункт 2</li>  
  <li>Пункт 3</li>  
</ul>
```

Списки — нумерованный

```
<ul type="1 или i или I или a или A" reversed start="число">  
  <li>Пункт 1</li>  
  <li>Пункт 2</li>  
  <li>Пункт 3</li>  
</ul>
```

Теги для форматирования текста

```
<em>Текст курсивом</em>
```

```
<strong>Жирный текст</strong>
```

```
<p>Первоначальное значение слова <strong>история</strong>  
восходит к древнегреческому термину, означавшему <em>расспрашивание,  
узнавание</em></p>
```

Цитаты

```
<blockquote cite="http://example.com/facts">  
  <p>Текст цитаты</p>  
</blockquote>
```

```
<blockquote cite="http://example.com/facts">  
  <p>Текст цитаты</p>  
  <p><em></em></p>  
</blockquote>
```

Атрибут `cite` не обязателен.

Список определений

```
<dl>
  <dt>Атом</dt>
  <dd>это наименьшая частица химического элемента,
  сохраняющая все его химические свойства.</dd>

  <dt>Молекула</dt>
  <dd>мельчайшая частица вещества (например, химического
  соединения), определяющая химические свойства этого
  вещества.</dd>
</dl>
```

Гиперссылки

```
<a href="http://yandex.ru">Yandex.ru</a>  
<a href="mailto:ivanovaev90@gmail.com">  
    ivanovaev90@gmail.com  
</a>  
<a href="#name">Внутренняя ссылка к заголовку (якорь)</a>  
  
<h1 id="name">Заголовок</h1>
```

Графические элементы

```

```

Определяем путь к файлу:

- _ — В папке где находится html, на одном уровне.
- ../ — Уходим на 1 уровень выше.
- ../.. — На 2 уровня выше и т.д.
- ./images** — В папке images, которая рядом с html
- ./images/people** — Можно уходить глубоко вниз

Разделители

`
` или `
`
`<hr />` или `<hr>`

Группировка заголовков

```
<hgroup>  
  <h1>Кристина Ветрова</h1>  
  <h2>Персональный сайт</h2>  
</hgroup>
```

Тег <section>

— Задаёт раздел документа, может применяться для блока новостей, контактной информации, глав текста, вкладок в диалоговом окне и др.

```
<section>
  <h1>Съемки фильма Полипропилен</h1>
  <p>История о том, как снимали фильм, где герои отдыхали
на пляже, потом пришел антагонист, избил протагонистов,
сбросил их в бассейн , и что из этого получилось.</p>
</section>
<section>
  <h1>Хорошиц язык</h1>
  <p>История о том, как проходила студия изучения языка
эсперато, в то время, как над ней, на веранде велась
студия приколистов, где травились анекдоты, и что из
этого получилось.</p>
</section>
```

Тег <article>

— Тег <article> задает содержание сайта вроде новости, статьи, записи блога, форума или др.

```
<article>
```

```
  История о том, как возле столовой появились загадочные  
  розовые следы с шестью пальцами, и почему это случилось.
```

```
</article>
```


Таблицы

```
<table>
  <caption>Заголовок таблицы</caption>
  <thead>
    <tr>
      <th>Заголовок ячейки 1</th>
      <th>Заголовок ячейки 2</th>
      <th>Заголовок ячейки 3</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Содержимое 1.1</td>
      <td>Содержимое 1.2</td>
      <td>Содержимое 1.3</td>
    </tr>
    <tr>
      <td>Содержимое 2.1</td>
      <td>Содержимое 2.2</td>
      <td>Содержимое 2.3</td>
    </tr>
  </tbody>
</table>
```

Шапка таблицы

tr задает строку в таблице

Ячейка с жирным текстом.

Тело таблицы

Ячейка с обычным текстом.

Таблицы

Численность населения

Годы	Все население, млн	Городское	Сельское
2010	142	105	37
2011	142	105	37
2012	143	105	38
2013	143	106	37
2014	143	106	37
2015	146	108	38
2016	145	108	37

Таблицы — атрибуты

```
<td colspan="2"></td>  
<th colspan="2"></th>
```

← Количество объединенных столбцов.

```
<td rowspan="2"></td>  
<th rowspan="2"></th>
```

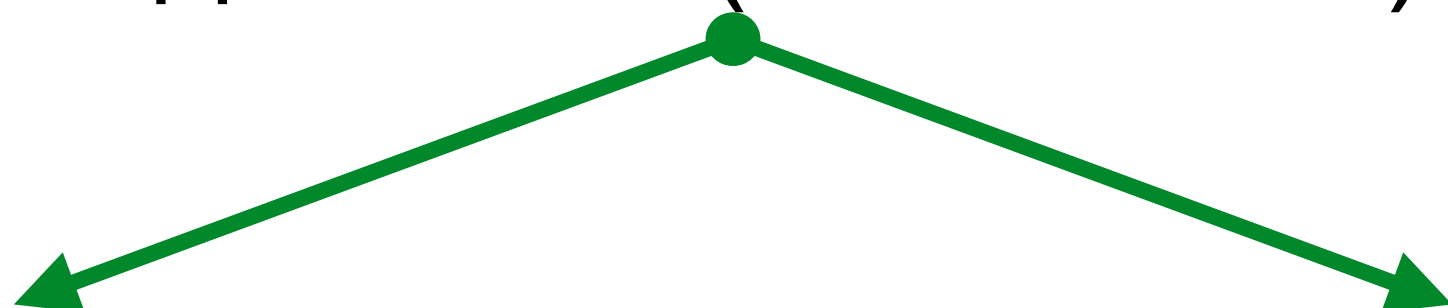
← Количество объединенных строк.

Усложненные таблицы

Численность населения

Годы	Все население, млн	В том числе:	
		Городское	Сельское
2010	142	105	37
2011	142	105	37
2012	143	105	38
2013	143	106	37
2014	143	106	37
2015	146	108	38
2016	145	108	37

Виды тегов (элементов)



Строчные

Являются
непосредственной частью
строки.

**em, strong, a, img,
span**

Блочные

Занимает всю доступную
ширину, высота элемента
определяется его
содержимым, и он всегда
начинается с новой
строки.

**h1, ..., h6, p, blockquote,
table, ol, ul, pre, hr, br,
div, header, nav, main,
footer**

Правила блочных элементов

- **Запрещено** вставлять блочный элемент внутрь строчного: `<p></p>`
- По **ширине блочные элементы** занимают **всё допустимое пространство**.
- **Блоки располагаются** по вертикали **друг под другом**.
- **Текст** по умолчанию **выравнивается по левому краю**.

Правила строчных элементов

- Внутри строчных элементов **допустимо** помещать **текст или другие строчные элементы**. Вставлять **блочные элементы** **внутри строчных запрещено**.
- Несколько **строчных элементов идущих подряд** **располагаются на одной строке** и переносятся на другую строку при необходимости.

Элементы оформления



The screenshot shows a code editor window titled "index.html — My.Artec3D.com". The editor contains the following HTML code:

```
1 <div>Обертка</div>
2 <span>Обертка</span>
3
4 |
5
```

Two red arrows point from text annotations to the code:

- An arrow points from the text "Обертка для блочных элементов" to the `<div>` tag on line 1.
- An arrow points from the text "Обертка для строчных элементов" to the `` tag on line 2.

Типичный вид веб-страницы

Шапка

Навигация

Основное содержимое страницы

Подвал

Дополнительные элементы обертки

`<header>...</header>`

`<nav>...</nav>`

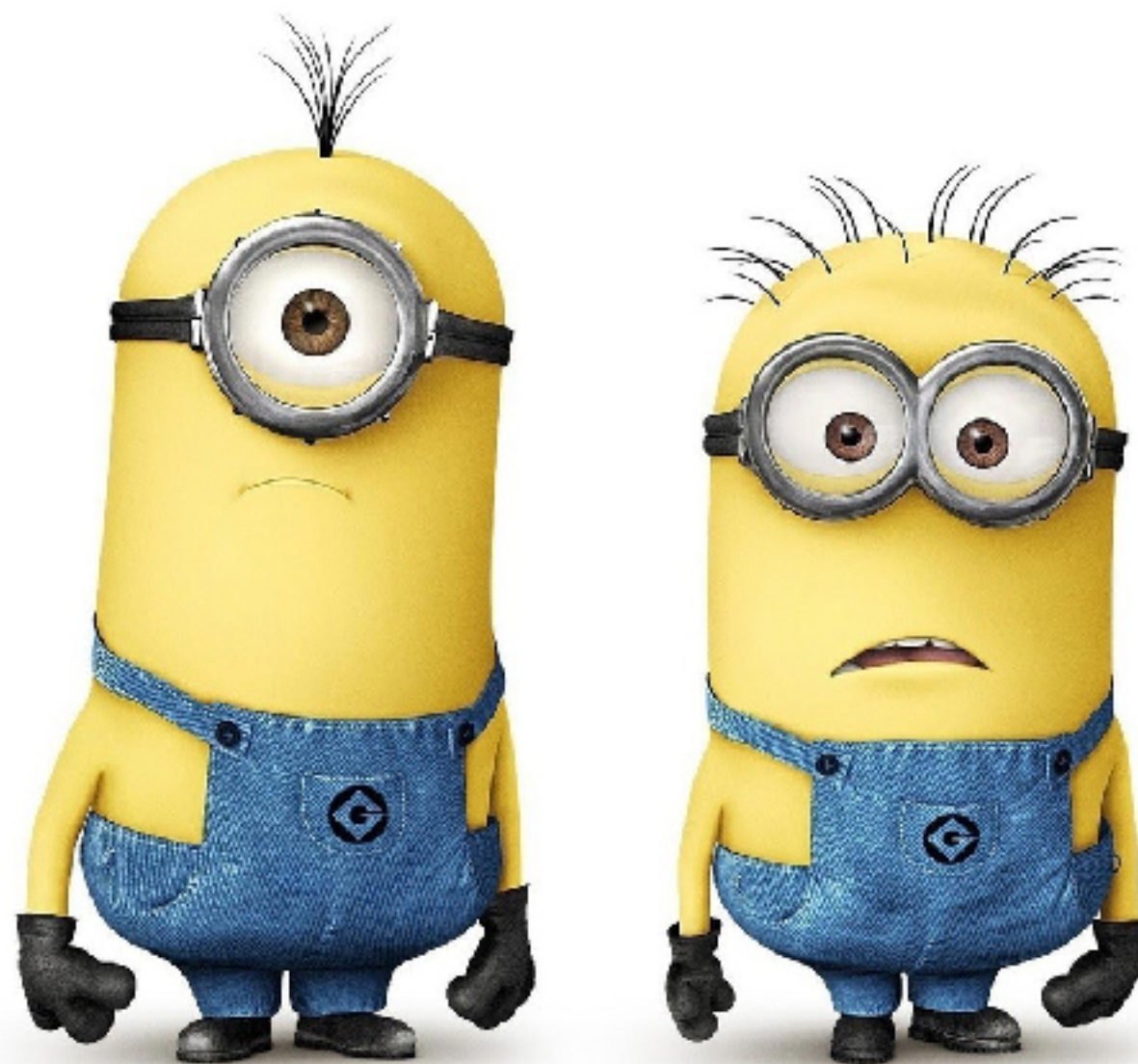
`<main>...</main>`

`<footer>...</footer>`

Семантика в HTML и почему это важно

Что такое семантика?

— наука о понимании определенных знаков, последовательностей символов и других условных обозначений.



Заголовок слайда

(Подзаголовок слайда, небольшая цитата)

Момент этот заключается в понимании такого важного понятия, как **семантика кода**. Давайте в этой заметке попытаемся разобраться с этим вопросом и *зачем это все нужно*.

Подзаголовок

- Пункт 1
- Пункт 2
- Пункт 3
- Пункт 4

Заголовок слайда (Подзаголовок слайда, небольшая цитата). Момент этот заключается в понимании такого важного понятия, как семантика кода. Давайте в этой заметке попытаемся разобраться с этим вопросом и зачем это все нужно. Подзаголовок. Пункт 1. Пункт 2. Пункт 3. Пункт 4. Заголовок слайда (Подзаголовок слайда, небольшая цитата). Момент этот заключается в понимании такого важного понятия, как семантика кода. Давайте в этой заметке попытаемся разобраться с этим вопросом и зачем это все нужно. Подзаголовок. Пункт 1. Пункт 2. Пункт 3. Пункт 4.

Семантика в HTML

Теги, созданы для того, чтобы делать документ понятным браузеру, чтобы ему было легко его читать!

Поэтому, **всегда думайте над содержимым**, какие теги использовать для отображения. Как бы вы оформили все на бумаге, сделайте это на html!

Преимущества семантикой верстки

1. Браузер без стилей сможет **корректно отображать ваш документ.**
2. Семантичный код **лучше понимается поисковыми системами.**
3. Оформленный код **более приятен человеку.**
4. **Легче поддерживать** код.

Практика

Начинаем работу!



Домашнее задание

Домашнее задание

1. Пройти в HTMLAcademy:
 - структуре HTML до подключения стилей.
 - разметка текста.
 - ссылки и изображения.
 - знакомство с таблицами. (Простейшая таблица, добавляем строки и столбцы, заголовки таблицы, объединение ячеек, закрепление colspan и rowspan).
 - знакомство с формами. (По желанию).

Вопросы?



Автор курса: Елена Иванова
lessons4.liveldi.ru