

ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Лекция 1

Введение в HTML. Теги и их атрибуты. Структура HTML-кода

Лектор: Берёза Евгений Олегович
bereza.evgenij@gmail.com

Компетенции

Общепрофессиональные компетенции

ОПК-1 способность использовать нормативные-правовые документы, международные и отечественные стандарты в области информационных технологий и систем и технологий в профессиональной деятельности

–ОПК-1.1 – способность использовать современные стандарты в области информационных технологий

Профессиональные компетенции

ПК-8 способность программировать приложения и создавать программные прототипы решения прикладных задач

–ПК-8.2 – программировать на языках сценариев

Тема 1. Содержание

HTML

1. Введение в HTML
2. Инструментарий
3. Теги
4. Структура HTML-кода
5. Типы тегов
6. Устаревшие теги

Тема 1. Введение в HTML

HTML (HyperText Markup Language, язык разметки гипертекста) – это стандартный язык разметки документов в Интернете, который определяет, как и какие элементы должны располагаться на веб-странице.

HTML не является языком программирования!

Текстовые документы, содержащие разметку на языке HTML (HTML-документы), имеют следующие расширения файлов:

.html

.htm

Специальные приложения, которые отображают HTML-документ в его форматированном виде, называются **«браузерами»**. Браузеры обычно предоставляют пользователю удобный интерфейс для запроса веб-страниц, их просмотра (и вывода на иные внешние устройства) и, при необходимости, отправки введённых пользователем данных на сервер.

Наиболее популярными на сегодняшний день браузерами являются **Google Chrome, Mozilla Firefox, Opera, Internet Explorer** и **Safari**.

Человек, создающий HTML-документы, обычно называется **«верстальщиком»**.

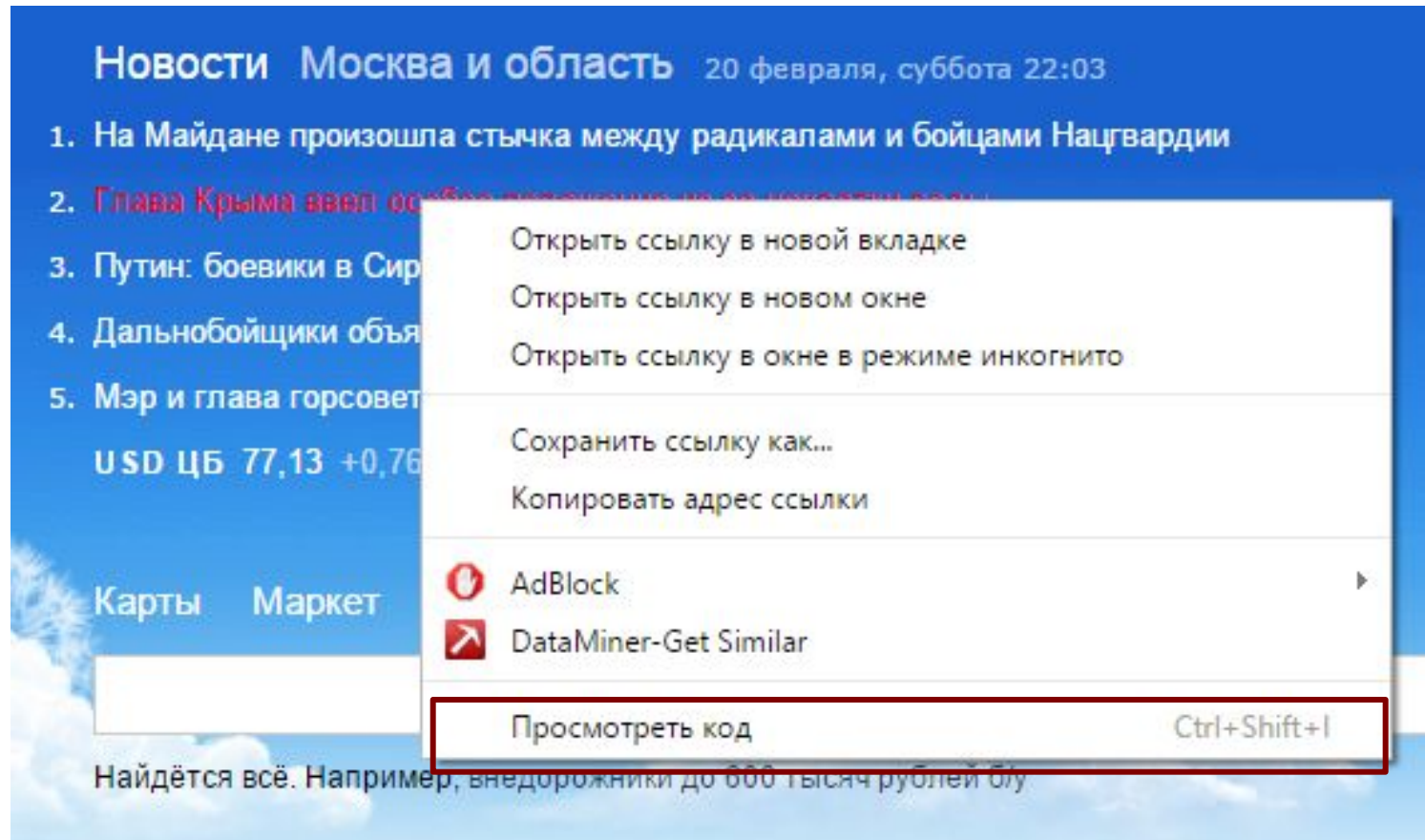
Тема 1. Инструментарий

При создании HTML-документов, верстальщики обычно используют следующий минимальный набор инструментов:

- 1.Текстовый редактор** (блокнот, PSPad, HtmlReader, Notepad++, EditPlus) или **WYSIWYG HTML-редактор** (Adobe Dreamweaver, Microsoft Visual Studio) или специализированная интегрированная среда разработки, по-другому называемые **IDE** – Integrated development environment (Visual Web Developer, WebStorm, Eclipse).
- 2.Браузер** для просмотра результатов (Google Chrome, Firefox, Opera, Safari и Internet Explorer) и **панель просмотра кода элемента**.
- 3.Валидатор** — программа для проверки синтаксиса HTML и выявления ошибок в коде.
- 4.Графический редактор** (Photoshop, Paint.Net, Paint).
- 5.Справочник по тегам HTML** (<http://www.w3.org/TR/html4/index/elements.html>, <https://webref.ru>, <http://htmlbook.ru>).

Тема 1. Инструментарий.

Панель просмотра кода элемента



Для доступа к панели просмотра кода элемента необходимо правой кнопкой нажать на элемент на странице браузера и в контекстном меню выбрать пункт «Просмотр кода элемента» для браузера Google Chrome или похожий пункт меню в других браузерах.

Тема 1. Инструментарий. Панель просмотра кода элемента

The screenshot shows a web browser interface with a news article and its developer tools. The news article is titled "Новости Москва и область" and dated "20 февраля, суббота 22:01". The main headline is "На Майдане произошла стычка между радикалами и бойцами Нацгвардии". A list of news items follows, with the second item highlighted: "Глава Крыма ввел особое положение из-за нехватки воды". The developer tools are open, showing the HTML structure of the page. The selected element is a link with the text "Глава Крыма ввел особое положение из-за нехватки воды". The HTML code for this link is: `Глава Крыма ввел особое положение из-за нехватки воды`. The developer tools also show the CSS styles for the selected element, including a color of #fff.

Сделать стартовой Солнечногорск

Настройка kereza.evgenij

Почта 1066

Деньги 0 Р

Диск

Написать письмо

Новости Москва и область 20 февраля, суббота 22:01

1. На Майдане произошла стычка между радикалами и бойцами Нацгвардии

2. Глава Крыма ввел особое положение из-за нехватки воды

3. a.link_list_item-content.link_black_yes 356px × 15px CHG

4. Дальнебойщики объявили забастовку в 45 регионах

5. Мэр и глава горсовета Бахчисарая подали в отставку из-за видео

USD ЦБ 77,13 +0,76 EUR ЦБ 85,86 +0,94 НЕФТЬ 33,12 -0,42% ...

Красивые фоны для вашего браузера

Элементы Яндекса

Карты Маркет Новости Переводчик Картинки Видео Музыка ещё

Найти

Elements Console Sources Network Timeline Profiles Resources Security Audits

```
<div class="content-tabs_head" role="tablist">...</div>
<div class="content-tabs_items content-tabs_items_active_true" role="tabpanel" id="tab_topnews-1_newsc" aria-labelledby="tab_topnews-1_news" aria-hidden="false">
  <ol class="list b-news-list">
    <li class="list_item">...</li>
    <li class="list_item">
      ::before
      <a href="https://news.yandex.ru/yandsearch?cl4url=www.ntv.ru/novosti/1607708/&lang=ru&lr=10755" class="link_list_item-content link_black_yes" aria-label="Глава Крыма ввел особое положение из-за нехватки воды">Глава Крыма ввел особое положение из-за нехватки воды</a>
    </li>
    <li class="list_item">...</li>
    <li class="list_item">...</li>
    <li class="list_item list_item_last">...</li>
  </ol>
</div>
<div class="content-tabs_items content-tabs_items_active_false" role="tabpanel" id="tab_topnews-1_regionc" aria-labelledby="...>
```

html body div div div div div div #wd-wrapper_topnews-1 #wd_topnews-1 div div div #tab_topnews-1_newsc ol li.list_item a.link_list_item-content.link_black_yes

Styles Computed Event Listeners >>

Filter +, #, ◆

```
element.style {
}
@p11Vb407HIMo-1_Ak@CEfYm.css:1
.container__first .link_black_yes,
.container__first
.link_black_yes.link_pseudo_yes,
.container__first .link_black_yes:link,
.container__first .link_black_yes:visited {
  color: #fff;
}
@p11Vb407HIMo-1_Ak@CEfYm.css:1
.container__first .link, .container__first
.link.link_pseudo_yes, .container__first
.link:link, .container__first .link:visited,
```

Тема 1. Теги

Одиночные теги:

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

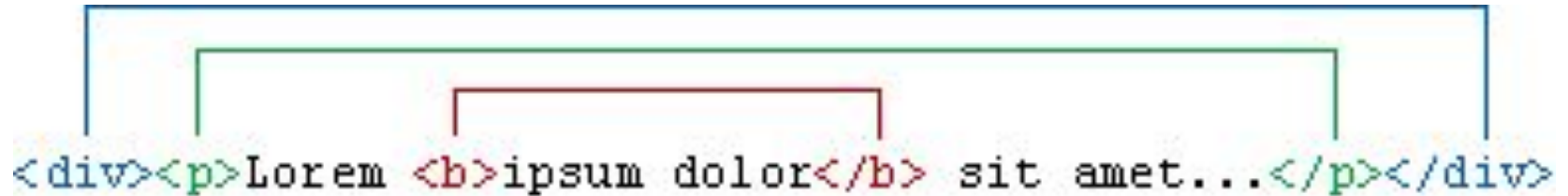
Парные теги (контейнеры):

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

```
1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2  <html>
3  <head>
4      <meta http-equiv="content-type" content="text/html; charset=utf-8">
5      <title>Заголовок браузера</title>
6  </head>
7  <body>
8      
9      <p>Текст</p>
10     <br>
11     <div>Блок <span>Текст внутри блока</span></div>
12 </body>
13 </html>
```

Рисунок 1. Назовите все одиночные и все парные теги?

Тема 1. Теги. Вложение тегов



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

Рисунок 1. Правильное вложение тегов



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

Рисунок 2. Ошибочное вложение тегов

Тема 1. Теги. Правила применения тегов

Атрибуты тегов и кавычки

Все значения атрибутов тегов следует указывать в двойных или одинарных кавычках.

```
1 |<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 |<html>
3 |<head>
4 |   <meta http-equiv="content-type" content="text/html; charset=utf-8">
5 |   <title>Кавычки в атрибуте alt</title>
6 |</head>
7 |<body>
8 |  <p></p>
9 |  <p></p>
10|</body>
11|</html>
```

Рисунок 1. Пример атрибутов и тегов

Регистр имен тегов

Теги можно писать как прописными, так и строчными символами (
,
 или
).

Тема 1. Теги. Правила применения тегов

Переносы строк между атрибутами тега

Внутри тега между его атрибутами допустимо ставить перенос строк.

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4     <meta http-equiv="content-type" content="text/html; charset=utf-8">
5     <title>Кавычки в атрибуте alt</title>
6 </head>
7 <body>
8 <p></p>
9 <p></p>
13 </body>
14 </html>
```

Рисунок 1. Переносы строк внутри атрибутов

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.

Тема 1. Теги. Правила применения тегов

Порядок тегов

Существует определенная иерархия вложенности тегов. Например, тег `<title>` должен находиться внутри контейнера `<head>` и нигде иначе.

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.

```
1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2  <html>
3  <head>
4      <meta http-equiv="content-type" content="text/html; charset=utf-8">
5      <title>Заголовок браузера</title>
6  </head>
7  <body>
8      
9      <p>Текст</p>
10     <br>
11     <div>Блок <span>Текст внутри блока</span></div>
12 </body>
13 </html>
```

Рисунок 1. Порядок тегов

Закрытие тегов

Существует три состояния закрывающего тега: обязателен, не требуется или не обязателен. Обязательный закрывающий тег должен присутствовать всегда, Рекомендуется также закрывать необязательные теги.

Тема 1. Теги. Атрибуты тегов

Значения по умолчанию для атрибутов

Если для тега не добавлен какой-либо допустимый атрибут, это означает, что браузер в этом случае будет подставлять значение, заданное по умолчанию. Если вы ожидали получить иной результат на веб-странице, проверьте, возможно, следует явно указать значения некоторых атрибутов.

Атрибуты без значений

Допустимо использовать некоторые атрибуты у тегов, не присваивая им никакого значения. Подобная запись называется «сокращенный атрибут тега».

```
1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2  <html>
3  <head>
4      <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5      <title>Добавление формы</title>
6  </head>
7  <body>
8      <form action="self.php">
9          <p><input type="text"></p>
10         <p><input type="submit" disabled></p>
11     </form>
12 </body>
13 </html>
```

Рисунок 1. Пример атрибута без значения (атрибут disabled)

Тема 1. Теги. Атрибуты тегов

Порядок атрибутов в тегах

Порядок атрибутов в любом теге не имеет значения и на результат отображения элемента не влияет.

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Порядок атрибутов тега</title>
6 </head>
7 <body>
8   
9   
10 </body>
11 </html>
```

Рисунок 1. Порядок атрибутов в тегах

Формат атрибутов

Каждый атрибут тега относится к определенному типу (например: текст, число, путь к файлу и др.), который обязательно должен учитываться при написании атрибута. Так, у тега ``, он добавляет на веб-страницу рисунок, а его атрибут `width` задает ширину изображения в пикселах. Если поставить не число, а нечто другое, то значение будет проигнорировано и возникнет ошибка при валидации документа.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<!DOCTYPE>

Предназначен для указания типа текущего документа — **DTD** (**document type definition**, описание типа документа). Это необходимо, чтобы браузер понимал, как следует интерпретировать текущую веб-страницу, ведь HTML существует в нескольких версиях, кроме того, имеется **XHTML** (**EXtensible HyperText Markup Language**, расширенный язык разметки гипертекста), похожий на HTML, но различающийся с ним по синтаксису. Чтобы браузер «не путался» и понимал, согласно какому стандарту отображать веб-страницу и необходимо в первой строке кода задавать <!DOCTYPE>

Тема 1. Структура HTML-кода. <!DOCTYPE>

DOCTYPE	Описание
HTML 4.01	
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	Строгий синтаксис HTML.
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	Переходный синтаксис HTML.
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">	В HTML-документе применяются фреймы.
HTML 5	
<!DOCTYPE html>	В этой версии HTML только один доктайп.
XHTML 1.0	
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">	Строгий синтаксис XHTML.
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">	Переходный синтаксис XHTML.
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">	Документ написан на XHTML и содержит фреймы.
XHTML 1.1	
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">	Разработчики XHTML 1.1 предполагают, что он постепенно вытеснит HTML. Как видите, никакого деления на виды это определение не имеет, поскольку синтаксис один и подчиняется четким правилам.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<html>

Определяет начало HTML-файла, внутри него хранится заголовок <head> и тело документа <body>

<head>

Заголовок документа, как еще называют блок <head>, может содержать текст и теги, но содержимое этого раздела не показывается напрямую на странице, за исключением контейнера <title>.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<title>

Определяет название вкладки веб-страницы, это один из важных элементов предназначенный для решения множества задач. Тег `<title>` является обязательным и должен непременно присутствовать в коде документа

</head>

Обязательно следует добавлять закрывающий тег `</head>`, чтобы показать, что блок заголовка документа завершен.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<meta>

Является универсальным и добавляет целый класс возможностей, в частности, с помощью метатегов, как обобщенно называют этот тег, можно изменять кодировку страницы, добавлять ключевые слова, описание документа и многое другое. Чтобы браузер понимал, что имеет дело с кодировкой UTF-8 (Unicode transformation format, формат преобразования Юникод) и добавляется строка в примере.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<body>

Тело документа, предназначено для размещения тегов и содержательной части веб-страницы.

<h1>

HTML предлагает шесть текстовых заголовков разного уровня, которые показывают относительную важность секции, расположенной после заголовка. Так, тег **<h1>** представляет собой наиболее важный заголовок первого уровня, а тег **<h6>** служит для обозначения заголовка шестого уровня и является наименее значительным.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<!-- Комментарий -->

Некоторый текст можно спрятать от показа в браузере, сделав его комментарием. Хотя такой текст пользователь не увидит, он все равно будет передаваться в документе, так что, посмотрев исходный код, можно обнаружить скрытые заметки.

Комментарии нужны для внесения в код своих записей, не влияющих на вид страницы. Начинаются они тегом `<!--` и заканчиваются тегом `-->`. Все, что находится между этими тегами, отображаться на веб-странице не будет.

Тема 1. Структура HTML-кода

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5   <title>Заголовок браузера</title>
6 </head>
7 <body>
8   <h1>Заголовок</h1>
9   <!-- Комментарий -->
10  <p>Первый абзац.</p>
11  <p>Второй абзац.</p>
12 </body>
13 </html>
```

Рисунок 1. Структура кода веб-страницы

<p>

Определяет абзац (параграф) текста. Если закрывающего тега нет, считается, что конец абзаца совпадает с началом следующего блочного элемента.

</body>

Следует добавить закрывающий тег </body>, чтобы показать, что тело документа завершено.

</html>

Последним элементом в коде всегда идет закрывающий тег </html>.

Тема 1. Типы тегов

Условно теги делятся на следующие типы:

- теги верхнего уровня;
- теги заголовка документа;
- блочные элементы;
- строчные элементы;
- универсальные элементы;
- списки;
- таблицы;
- фреймы.

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

Тема 1. Типы тегов. Теги верхнего уровня

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

<html></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 непременно их использование.

<head></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.

<body></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 относится текст, изображения, таблицы, списки и др.

Тема 1. Типы тегов. Теги верхнего уровня

```
1 <html>
2   <head>
3     | ...
4   </head>
5   <body>
6     ...
7   </body>
8 </html>
```

Контейнер `<html>` определяет «каркас» всей веб-страницы, внутри него вначале задается тег `<head>`, затем идет контейнер `<body>`, в нем хранится содержательная часть документа, которая и отображается в браузере. Теги `<html>` и `<body>` хотя и не относятся к обязательным тегам (т. е. их можно не размещать в коде), все же стоит добавлять всегда. Это позволяет получить четкую и понятную структуру документа.

Заметьте, что в примере не упоминается `<!DOCTYPE>`, поскольку этот обязательный элемент кода веб-страницы не является тегом, а предназначен для браузеров, чтобы сообщить им, как интерпретировать текущий документ.

Тема 1. Типы тегов. 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></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.

`<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.

Тема 1. Типы тегов. Теги заголовка документа

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4     <title>HTML</title>
5     <meta name="description" content="Сайт об HTML и создании сайтов">
6     <meta http-equiv="content-type" content="text/html; charset=utf-8">
7 </head>
8 <body>
9     <p>...</p>
10 </body>
11 </html>
```

Для краткого описания содержимого веб-страницы используется значение **description** атрибута **name**. Описание сайта, заданное с помощью тега `<meta>` и значения `description`, обычно отображается в поисковых **системах** или **каталогах** при **выводе результатов поиска**

Тема 1. Типы тегов. Теги заголовка документа

```
1 |<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 |<html>
3 |<head>
4 |   <title>HTML</title>
5 |   <meta name="keywords" content="HTML, META, метатег, тег, поисковая система">
6 |   <meta http-equiv="content-type" content="text/html; charset=utf-8">
7 |</head>
8 |<body>
9 |  <p>...</p>
10|</body>
11|</html>
```

Значение **keywords** также предназначено в первую очередь для повышения рейтинга сайта в поисковых системах, в нем перечисляются ключевые слова, встречаемые на веб-странице. Ключевые слова можно перечислять через пробел или запятую. Поисковые системы сами приведут запись к виду, который они используют.

Тема 1. Типы тегов. Блочные элементы

Блочные элементы характеризуются тем, что занимают всю доступную ширину, высота элемента определяется его содержимым, и он всегда начинается с новой строки.

<blockquote></blockquote>

Предназначен для выделения длинных цитат внутри документа. Текст, обозначенный этим тегом, традиционно отображается как выровненный блок с отступами слева и справа (примерно по 40 пикселей), а также с пустым пространством сверху и снизу.

<div></div>

Относится к универсальным блочным контейнерам и применяется в тех случаях, где нужны блочные элементы без дополнительных свойств. Также с помощью тега <div> можно выравнивать текст внутри этого контейнера с помощью атрибута align.

<h1></h1>, ..., <h6></h6>

Эта группа тегов определяет текстовые заголовки разного уровня, которые показывают относительную важность секции, расположенной после заголовка.

Тема 1. Типы тегов. Блочные элементы

<hr>

Рисует горизонтальную линию, которая по своему виду зависит от используемых атрибутов. Линия всегда начинается с новой строки, а после нее все элементы отображаются на следующей строке.

<p></p>

Определяет параграф (абзац) текста.

<pre></pre>

Задаёт блок предварительно форматированного текста. Такой текст отображается обычно моноширинным шрифтом и со всеми пробелами между словами. В HTML любое количество пробелов идущих в коде подряд на веб-странице показывается как один. Тег **<pre>** позволяет обойти эту особенность и отображать текст как требуется разработчику. Обычно тег **<pre>** используется для вставки исходного кода с сохранением его форматирования.

Следующие теги не должны размещаться внутри контейнера **<pre>**: ****, **** и ****.

Тема 1. Типы тегов. Строчные элементы

Строчными называются такие элементы веб-страницы, которые являются непосредственной частью другого элемента, например, текстового абзаца. В основном они используются для изменения вида текста или его логического выделения.

<a>

Тег <a> является одним из важных элементов HTML и предназначен для создания ссылок. В зависимости от присутствия атрибутов name или href тег <a> устанавливает ссылку или якорь.

Определяет жирное начертание шрифта.

**
**

Тег
 устанавливает перевод строки в том месте, где этот тег находится. В отличие от тега параграфа <p>, использование тега
 не добавляет пустой отступ перед строкой.

Тема 1. Типы тегов. Строчные элементы

Предназначен для акцентирования текста. Браузеры отображают такой текст курсивным начертанием.

<i></i>

Устанавливает курсивное начертание шрифта.

Предназначен для отображения на веб-странице изображений в графическом формате GIF, JPEG или PNG. Если необходимо, то рисунок можно сделать ссылкой на другой файл, поместив тег `` в контейнер `<a>`. При этом вокруг изображения отображается рамка, которую можно убрать, добавив атрибут `border="0"` в тег ``.

Универсальный тег, предназначенный для определения строчного элемента внутри документа.

Тема 1. Типы тегов. Строчные элементы

Отображает шрифт в виде нижнего индекса. Текст при этом располагается ниже базовой линии остальных символов строки и уменьшенного размера — H₂O.

Отображает шрифт в виде верхнего индекса. По своему действию похож на <sub>, но текст отображается выше базовой линии текста — м².

Тема 1. Типы тегов.

Разница между блочными и строчными элементами

1. **Строчные элементы** могут содержать только данные или другие строчные элементы, а **в блочные допустимо вкладывать другие блочные элементы**, строчные элементы, а также данные. Иными словами, **строчные элементы никак не могут хранить блочные элементы.**
2. **Блочные элементы всегда начинаются с новой строки**, а строчные таким способом не акцентируются.
3. **Блочные элементы занимают всю доступную ширину**, например, окна браузера, а **ширина строчных элементов равна их содержимому плюс значения отступов, полей и границ.**

Тема 1. Типы тегов. Универсальные элементы

Особенность этих тегов состоит в том, что они в зависимости от контекста могут использоваться как блочные или встроенные элементы.

Используется для выделения текста, который был удален в новой версии документа. Подобное форматирование позволяет отследить, какие изменения в тексте документа были сделаны. Браузеры обычно помечают текст в контейнере `` как перечеркнутый.

<ins>

Предназначен для акцентирования вновь добавленного текста и обычно применяется наряду с тегом ``. Браузеры помечают содержимое контейнера `<ins>` подчеркиванием текста.

Тема 1. Типы тегов. Теги для списков

Списком называется взаимосвязанный набор отдельных фраз или предложений, которые начинаются с маркера или цифры. Списки предоставляют возможность упорядочить и систематизировать разные данные и представить их в наглядном и удобном для пользователя виде.

Тег устанавливает нумерованный список, т.е. каждый элемент списка начинается с числа или буквы и увеличивается по нарастающей.

Устанавливает маркированный список, каждый элемент которого начинается с небольшого символа — маркера.

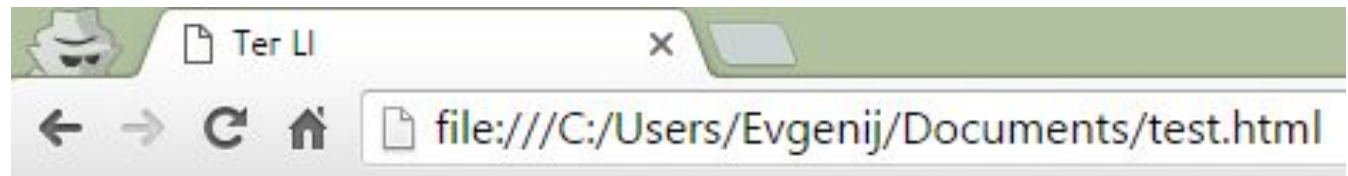
Тег определяет отдельный элемент списка. Внешний тег или устанавливает тип списка — маркированный или нумерованный.

<dd>, <dt>, <dl>

Тройка элементов предназначена для создания списка определений. Каждый такой список начинается с контейнера <dl>, куда входит тег <dt> создающий термин и тег <dd> задающий определение этого термина. Закрывающий тег </dd> не обязателен, поскольку следующий тег сообщает о завершении предыдущего элемента. Тем не менее, хорошим стилем является закрывать все теги.

Тема 1. Типы тегов. Теги для таблиц

```
1 |<!DOCTYPE HTML>
2 |<html>
3 |<head>
4 |   <meta charset="utf-8">
5 |   <title>Ter LI</title>
6 |</head>
7 |<body>
8 |<ul>
9 |   <li>Чебурашка</li>
10|   <li>Крокодил Гена</li>
11|   <li>Шапокляк</li>
12|</ul>
13|</body>
14|</html>
```



- Чебурашка
- Крокодил Гена
- Шапокляк

Тема 1. Типы тегов. Теги для таблиц

Таблица состоит из строк и столбцов ячеек, которые могут содержать текст и рисунки. Обычно таблицы используются для упорядочения и представления табличных данных.

<table>

Служит контейнером для элементов, определяющих содержимое таблицы. Любая таблица состоит из строк и ячеек, которые задаются с помощью тегов `<tr>` и `<td>`.

<tr>

Тег `<tr>` служит контейнером для создания строки таблицы.

<th>

Тег `<th>` предназначен для создания одной ячейки таблицы, которая обозначается как заголовочная. Текст в такой ячейке отображается браузером обычно жирным шрифтом и выравнивается по центру.

<td>

Предназначен для создания одной ячейки таблицы. Тег `<td>` должен размещаться внутри контейнера `<tr>`, который в свою очередь располагается внутри тега `<table>`.

Тема 1. Типы тегов. Теги для таблиц

```
1 |<!DOCTYPE HTML>
2 |<html>
3 |<head>
4 |   <meta charset="utf-8">
5 |   <title>Таблица размеров обуви</title>
6 |</head>
7 |<body>
8 |<table border="1" cellpadding="0px" cellspacing="0px">
9 |   <caption>Таблица размеров обуви</caption>
10 |   <tr>
11 |     <th>Россия</th>
12 |     <th>Великобритания</th>
13 |     <th>Европа</th>
14 |     <th>Длина ступни, см</th>
15 |   </tr>
16 |   <tr>
17 |     <td>34,5</td>
18 |     <td>3,5</td>
19 |     <td>36</td>
20 |     <td>23</td>
21 |   </tr>
22 |   <tr>
23 |     <td>35,5</td>
24 |     <td>4</td>
25 |     <td>36½</td>
26 |     <td>23-23,5</td>
27 |   </tr>
28 | </table>
29 | </body>
30 | </html>
```

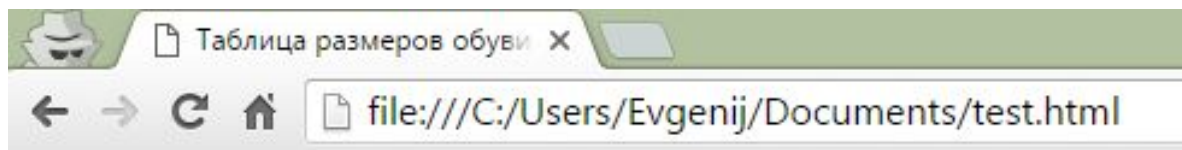


Таблица размеров обуви

Россия	Великобритания	Европа	Длина ступни, см
34,5	3,5	36	23
35,5	4	36½	23-23,5

Тема 1. Типы тегов. Теги для фреймов

Таблица состоит из строк и столбцов ячеек, которые могут содержать текст и рисунки. Обычно таблицы используются для упорядочения и представления табличных данных.

<iframe></iframe>

Создает плавающий фрейм, который находится внутри обычного документа, он позволяет загружать в область заданных размеров любые другие независимые документы.

Тег **<iframe>** является контейнером, содержание которого игнорируется браузерами, не поддерживающими данный тег. Для таких браузеров можно указать альтернативный текст, который увидят пользователи. Он должен располагаться между элементами **<iframe>** и **</iframe>**.

Тема 1. Значения атрибутов тегов

Атрибуты тегов расширяют возможности самих тегов и позволяют гибко управлять различными настройками отображения элементов веб-страницы. Общее количество атрибутов достаточно велико, но их значения, как правило, можно сгруппировать по разным типам, например, задающих цвет, размер, адрес и др.

Основные типы значений:

- цвет
- размер;
- адрес.

Тема 1. Значения атрибутов тегов. Цвет

В HTML цвет задается одним из двух путей: с помощью шестнадцатеричного кода и по названию некоторых цветов. Преимущественно используется способ, основанный на шестнадцатеричной системе исчисления, как наиболее универсальный.

Шестнадцатеричные цвета

Для задания цветов в HTML используются числа в шестнадцатеричном коде. Шестнадцатеричная система, в отличие от десятичной системы, базируется, как следует из ее названия, на числе 16. Цифры будут следующие: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Числа от 10 до 15 заменены латинскими буквами.

Тема 1. Устаревшие теги

<applet>	Добавляет Java-апплет в документ. Вместо него следует использовать <embed> или <object>
<acronym>	Этот тег вызывал постоянные вопросы, что такое акроним и чем он отличается от аббревиатуры. Для упрощения остался единственный тег <abbr>
<bgsound>	Определяет музыкальный файл, который будет проигрываться на веб-странице при её открытии. Для воспроизведения музыки используйте новый элемент <audio>
<dir>	Создает список, содержащий названия директорий, вместо него используйте
<frame>, <frameset>, <noframe>	Фреймы более не поддерживаются. Если они вам требуются, используйте другую версию HTML или <iframe> совместно со стилями
<isindex>	Предназначен для поискового индекса в текущем документе. Комбинация <form> и <input> лучше справляется с этой задачей
<listing>, <xmp>	Для вывода листинга программы предназначены <pre> и <code>

Тема 1. Устаревшие теги

<nextid>	Этот тег не предназначен для людей и указывает идентификатор следующего документа для автоматических редакторов HTML. Полностью исключён
<noembed>	Предназначен для отображения информации на веб-странице, если браузер не поддерживает работу с плагинами. В качестве альтернативы используйте <object>
<plaintext>	Отображает содержимое контейнера «как есть», любые теги выводятся как текст. Вместо тега используйте MIME-тип text/plain
<rb>	Определяет базовый текст внутри <ruby>. Этот тег полностью исключён
<strike>	Для зачёркнутого текста применяется <s>, а для указания редакторской правки
<basefont>, <big>, <blink>, <center>, , <marquee>, <multicol>, <nobr>, <spacer>, <tt>, <u>	Вместо этих тегов управляющих видом текста применяются стили