

Элементы технологии HTML

Артамонов Ю.Н.

Общее представление о языке разметки данных html

HTML (HyperText Markup Language — «язык гипертекстовой разметки») - это стандартизированный язык разметки документов в интернете. Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства. HTML-страницы, как правило, передаются браузерам от сервера по протоколам HTTP или HTTPS, в виде простого текста или с использованием шифрования.

Версии языка:

- HTML 2.0 — 24 ноября 1995 года;
- HTML 3.0 — 28 марта 1995 года — до 28 сентября 1995 года;
- HTML 3.2 — 14 января 1997 года;
- HTML 4.0 — 18 декабря 1997 года;
- HTML 4.01 — 24 декабря 1999 года;
- ISO/IEC 15445:2000 (так называемый ISO HTML, основан на HTML 4.01 Strict) — 15 мая 2000 года;
- HTML5 — 28 октября 2014 года;
- HTML 5.1 начал разрабатываться 17 декабря 2012 года. Рекомендован к применению с 1 ноября 2016 года;
- HTML 5.2 был представлен 14 декабря 2017 года.

Структура html документа

Любой документ на языке HTML представляет собой набор элементов, причём начало и конец каждого элемента обозначается специальными пометками — тегами. Тег — элемент языка разметки гипертекста. Текст, содержащийся между начальным и конечным тегом, отображается и размещается в соответствии со свойствами, указанными в начальном теге.

Текст может быть пустым, то есть не содержащим никакого текста и других данных. В этом случае обычно не указывается закрывающий тег (например, тег переноса строки `
` - одиночный и закрывать его не нужно), в противном случае у каждого начального тега есть конечный. Кроме того, элементы могут иметь атрибуты, определяющие какие-либо их свойства (например, атрибут `href=` у ссылки). Атрибуты указываются в открывающем теге. Вот примеры фрагментов HTML-документа:

```
<strong>Текст между двумя тегами — открывающим и закрывающим.</strong>
```

```
<a href="http://www.example.com">Здесь элемент содержит атрибут href, то  
есть гиперссылку.</a>
```

А вот пример пустого элемента: `
`

Регистр, в котором набрано имя элемента и имена атрибутов, в HTML значения не имеет.

Общая структура html документа

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>HTML Document</title>
  </head>
  <body>
    <p>
      <b>
        Этот текст будет полужирным , <i>а этот — ещё и курсивным</i>.
      </b>
    </p>
  </body>
</html>
```

Наберите данный текст в обычном текстовом редакторе (блокнот, етас и т.д.), сохраните с расширением html и откройте в браузере.

Каждый HTML-документ, отвечающий спецификации HTML какой-либо версии, должен начинаться со строки объявления версии HTML `<!DOCTYPE...>`, которая обычно выглядит примерно так:

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

Если эта строка не указана, то добиться корректного отображения документа в браузере становится труднее (хотя, как правило, большинство браузеров сами ее добавляют, если она отсутствует).

Далее обозначается начало и конец документа тегами `<html>` и `</html>` соответственно. Внутри этих тегов должны находиться теги заголовка (`<head></head>`) и тела (`<body></body>`) документа.

Элемент структуры DOCTYPE

Варианты DOCTYPE для HTML 4.01

Строгий: не содержит элементов, помеченных как «устаревшие» или «не одобряемые».

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

Переходный: содержит устаревшие теги в целях совместимости и упрощения перехода со старых версий HTML.

```
<!DOCTYPE HTML PUBLIC "-//W3C/DTDHTML 4.01 Transitional/EN" "http://www.w3.org/TR/html4/loose.dtd">
```

С фреймами: аналогичен переходному, но содержит также теги для создания наборов фреймов.

```
<!DOCTYPE HTML PUBLIC "-//W3C/DTDHTML 4.01 Frameset/EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

Варианты DOCTYPE для HTML 5

В HTML 5 используется только один вариант DOCTYPE:

```
<!DOCTYPE html>
```

Элемент структуры 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. Содержимое тега `<head>` не отображается напрямую на веб-странице, за исключением тега `<title>` устанавливающего заголовок окна веб-страницы.

Внутри контейнера `<head>` допускается размещать следующие элементы: `<base>`, `<basefont>`, `<bgsound>`, `<link>`, `<meta>`, `<script>`, `<style>`, `<title>`. Синтаксис

```
<head>
```

```
...
```

```
</head>
```

Закрывающий тег не обязателен.

Примеры использования элемента META

Указание кодировки документа:

```
<meta http-equiv="Content-Type" content="text/html";  
charset=windows-1251">
```

Информация об авторе:

```
<meta name="Author" content="Yuriy Artamonov">
```

Ключевые слова:

```
<meta name="keywords" content="hypertext technology, html">
```

Описание сайта:

```
<meta name="description" content="Мой сайт">
```


Еще один пример использования элемента META

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    Это основное содержимое страницы.
  </body>
</html>
```

Ну, а теперь, научившись создавать простейшие страницы, мы перейдем к изучению полезных тегов, которые следует использовать внутри тега `<body>`.

Тег абзаца p, разрыва строки br

Каждый абзац начинается с новой строки и имеет так называемую красную строку (это когда первая строка текста абзаца немного отступает вправо). По умолчанию красной строки нет, но ее можно сделать. Для разрыва строки следует написать тег `
` - этот тег особенный и не имеет закрывающего.

Абзац создается с помощью тега `<p>` таким образом:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <p>
      Это абзац с разрывом <b>br</b>строки
    </p>
  </body>
</html>
```

Теги заголовков h1, h2, h3, h4, h5, h6

Заголовки создаются с помощью тегов <h1>, <h2>, <h3>, <h4>, <h5>, <h6>. Они имеют разную степень важности. В заголовке h1 следует располагать название всей HTML страницы, в h2 - название блоков страницы, в h3 - название подблоков и так далее. Все заголовки по умолчанию жирные и имеют разный размер.

```
<body>
```

```
  <h1>Заголовок h1</h1>
```

```
  <h2>Заголовок h2</h2>
```

```
  <h3>Заголовок h3</h3>
```

```
  <h4>Заголовок h4</h4>
```

```
  <h5>Заголовок h5</h5>
```

```
  <h6>Заголовок h6</h6>
```

```
  <p>Это первый абзац.</p>
```

```
  <p>Это второй абзац.</p>
```

```
  <p>Это третий абзац.</p>
```

```
</body>
```

Тег жирного начертания

Можно сделать жирным и обычный текст - достаточно взять его в тег ``.

```
!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <p>
      Это обычный текст , а это <b>жирный</b> текст .
    </p>
  </body>
</html>
```

Тег курсивного начертания

Кроме жирного можно сделать также и курсив с помощью тега `<i>`:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <p>
      Это обычный текст , а это <i>курсивный</i> текст .
    </p>
  </body>
</html>
```

Теги списков

Списки создаются с помощью тега ``, внутри которого обязательно должны идти теги ``. Тег `ul` задает сам список, а в теги `li` следует поместить пункты списка (то есть одному `li` соответствует один маркер списка). Отдельно теги `li` использовать нельзя.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <ul>
      <li>Первый пункт списка.</li>
      <li>Второй пункт списка.</li>
      <li>Третий пункт списка.</li>
    </ul>
  </body>
</html>
```

Теги списков

Списки, созданные через тег `ul`, называются неупорядоченными списками. Существуют еще и упорядоченные списки, у которых вместо маркеров-точек будут маркеры-цифры. Такие списки вместо тега `ul` имеют тег ``, а пункты таких списков также создаются через теги `li`.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <ol>
      <li>Первый пункт списка.</li>
      <li>Второй пункт списка.</li>
      <li>Третий пункт списка.</li>
    </ol>
  </body>
</html>
```

Ссылка создается с помощью тега `<a>`, при этом у нее должен быть обязательный атрибут `href`, в котором хранится адрес той страницы, на которую ведет ссылка.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <a href="https://mguu.ru/">Ссылка на сайт вуза.</a>
  </body>
</html>
```


Тег картинок

Для вставки картинки предназначен тег ``, имеющий обязательный атрибут `src`, в котором хранится путь к файлу картинки. Тег `` не требует закрывающего тега.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    
  </body>
</html>
```



Тег картинок

Ссылкой может быть не только текст, но и картинка - для этого достаточно тег `` вложить в тег `<a>`.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <a href="https://mguu.ru/"></a>
  </body>
</html>
```

Комментарии в html

Комментарии в HTML оформляются следующим образом: сначала уголок, знак ! и два дефиса - `<!--`, потом текст комментария, а потом два дефиса и уголок - `-->`.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Это заголовок тайтл</title>
  </head>
  <body>
    <!--
      Это комментарий HTML —
      на странице сайта он не отображается.
    -->

    Это основное содержимое страницы.
  </body>
</html>
```

Таблицы в html

Главным является тег `<table>`, внутри которого должны лежать теги `<tr>`, которые создают ряды (строки) таблицы, а внутри них - теги `<td>`, которые создают ячейки.

```
<table border="1">
  <!--Это будет первый ряд таблицы:-->
  <tr>
    <td>Ячейка 1</td>
    <td>Ячейка 2</td>
  </tr>
  <!--Это будет второй ряд таблицы:-->
  <tr>
    <td>Ячейка 3</td>
    <td>Ячейка 4</td>
  </tr>
  <!--Это будет третий ряд таблицы:-->
  <tr>
    <td>Ячейка 5</td>
    <td>Ячейка 6</td>
  </tr>
</table>
```

Ячейки-заголовки в таблице

Кроме тегов `td` существуют также теги `<th>`, которые также создают заголовки таблицы.

```
<table border="1">
  <tr>
    <th>Имя</th>
    <th>Фамилия</th>
    <th>Зарплата</th>
  </tr>
  <tr>
    <td>Иван</td>
    <td>Иванов</td>
    <td>200$</td>
  </tr>
  <tr>
    <td>Николай</td>
    <td>Сидоров</td>
    <td>1000$</td>
  </tr>
</table>
```

Атрибут cellpadding в таблице

Расстояние между ячейками регулируется атрибутом cellpadding

```
<table border="1" cellpadding="10">  
  <tr>  
    <th>Имя</th>  
    <th>Фамилия</th>  
    <th>Зарплата</th>  
  </tr>  
  <tr>  
    <td>Иван</td>  
    <td>Иванов</td>  
    <td>200$</td>  
  </tr>  
  <tr>  
    <td>Николай</td>  
    <td>Сидоров</td>  
    <td>1000$</td>  
  </tr>  
</table>
```

Атрибут cellpadding в таблице

Атрибут cellpadding задает отступ между текстом и границей ячейки.

```
<table border="1" cellpadding="20">
  <tr>
    <th>Имя</th>
    <th>Фамилия</th>
    <th>Зарплата</th>
  </tr>
  <tr>
    <td>Иван</td>
    <td>Иванов</td>
    <td>200$</td>
  </tr>
  <tr>
    <td>Николай</td>
    <td>Сидоров</td>
    <td>1000$</td>
  </tr>
</table>
```

Ширина и высота в таблице

Существуют также атрибуты width и height, которые позволяют задать таблице ширину и высоту.

```
<table border="1" cellpadding="0" cellspacing="0" width="300" height="300">
  <tr>
    <th>Имя</th>
    <th>Фамилия</th>
    <th>Зарплата</th>
  </tr>
  <tr>
    <td>Иван</td>
    <td>Иванов</td>
    <td>200$</td>
  </tr>
  <tr>
    <td>Николай</td>
    <td>Сидоров</td>
    <td>1000$</td>
  </tr>
</table>
```