

HTML

Язык описания Web-страниц

По материалам курса University of
Washington

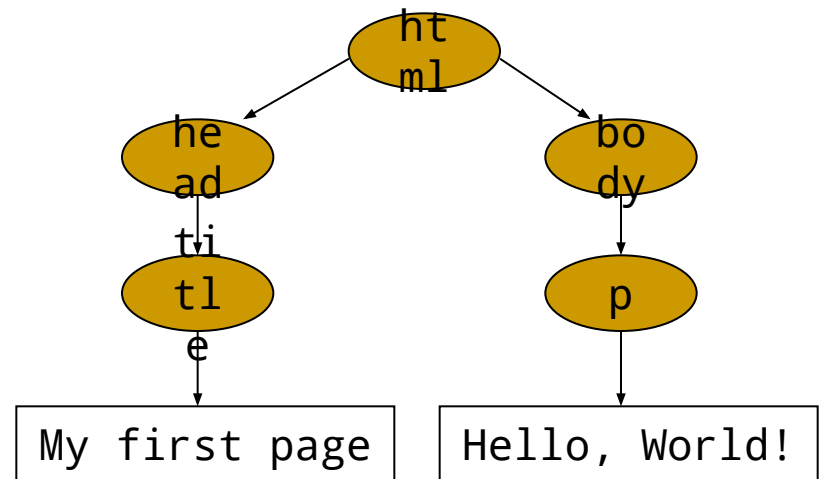
<http://www.cs.washington.edu/education/courses/cse190m/07sp/index.shtml>

Что такое HTML?

- это не язык программирования, а «язык разметки» (HyperText Markup Language);
- определяет содержание и структуру страницы, но не внешний вид;
- элементы языка имеют структуру дерева (вложенные элементы);
- пробелы либо игнорируются, либо заменяются одним пробелом;
- узлы дерева представляют собой либо текст (содержание), либо «структурные элементы», маркируемые «тегами» и имеющие «атрибуты».

Текст на языке HTML и его структура

```
<html>
  <head>
    <title>My first page</title>
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```



Структура элементов языка

Элемент, маркированный тегом, имеет следующую структуру:

```
<element attr1="value1" attr2="value2"...>  
    внутреннее содержание  
</element>
```

```
<a href="page2.html">Next page</a>
```

Если тег не имеет *внутреннего содержания*, то он может иметь упрощенную структуру:

```
<element attr1="value1" attr2="value2"... />
```

```

```

```
<hr/>
```

Структура страницы

Правильная страница имеет две части – заголовок (информация о странице) и тело (содержание страницы)

```
<html>
  <head>
    информация о странице
  </head>
  <body>
    внутреннее содержание
  </body>
</html>
```

Пример:

```
<html>
  <head>
    <title>My first web page</title>
  </head>
  <body>
    <h1>Welcome to my first web page!</h1>
    <p>This is a paragraph of text.</p>
  </body>
</html>
```

ХНТМЛ – более современный HTML (2000 год)

HTML + XML (eXtended Markup Language) = ХНТМЛ

Браузеры отображают пришедшую к ним по запросу страницу. Страница может быть написана как на «старом» HTML, так и на более новых XML и ХНТМЛ. ХНТМЛ -

- Более строгий и хорошо структурированный язык
- Лучшая «переносимость» между различными браузерами
- ХНТМЛ допускает вставки на других «языках разметки» XML, SVG, MathML, MusicML и т.д.

Структура документа в формате XHTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    информация о странице
  </head>
  <body>
    внутреннее содержание
  </body>
</html>
```

Основное отличие во внутреннем содержании документов HTML и XHTML: если браузер видит ошибку в HTML-документе, то он обязан постараться понять, что имел в виду автор документа. Если ошибка обнаружена в XHTML документе, то браузер просто сообщает об ошибке.

Кроме того, форматирование элементов XHTML-документа должно быть оформлено с помощью CSS-стилей. Многие теги, такие как `font` и атрибуты, такие как `bgcolor` и `align` не поддерживаются в XHTML.

Ошибки XHTML, которые допустимы в HTML

1. Все элементы должны быть закрыты.
2. Все обязательные атрибуты должны присутствовать.
3. Вложенность элементов должна быть правильной.
4. В теле документа текст не может быть вложен непосредственно.
5. Атрибуты должны заключаться в кавычки.
6. «Блочные» элементы не могут быть вложены в «строчные».
7. Спецсимволы всегда должны быть представлены мнемоническими ссылками.
8. Теги и атрибуты записываются только строчными буквами.

Неправильно :

1. `<hr>`
2. ``
3. `<i>some text</i>`
4. `<body>Hello, World!</body>`
5. ``
6. `<a><div> ... </div>`
7. `&`
8. ``

Правильно :

1. `<hr/>` (или `<hr> </hr>`)
2. ``
3. `<i>some text</i>`
4. `<body><p>Hello, World!</p></body>`
5. ``
6. `&`
7. `&`
8. ``

Блочные и строчные элементы

Блочные элементы содержат фрагменты текста, которые всегда отображаются в отдельных блоках. Браузеры всегда визуальнo отделяют блочные элементы друг от друга.

Примеры: `<div>`, `<p>`, ``, `<tr>`

Строчные элементы могут располагаться друг за другом в пределах одной строки

Примеры: ``, `<a>`, ``, ``

Комментарии вставляются в HTML текст так же, как и обычные элементы:
`<!-- Это моя самая прикольная страница -->`

Некоторые элементы HTML и их атрибуты

Абзац (параграф) – блочный элемент

`<p>`Видимое здесь содержимое
отображается в браузере в виде одного абзаца.
Повторяющиеся пробелы и переходы со строки на
строку игнорируются.`</p>`

Видимое здесь содержимое отображается в браузере в виде одного абзаца.
Повторяющиеся пробелы и переходы со строки на строку игнорируются.

Если необходимо сделать переход на следующую строку внутри абзаца,
то это достигается с помощью элемента `
`

`<p>`Это первая строка параграфа,`
`а это уже вторая`</p>`
`<p>`А вот это уже следующий абзац.`</p>`

Это первая строка параграфа,
а это уже вторая
А вот это уже следующий абзац.

Некоторые элементы HTML и их атрибуты

Заголовки – блочные элементы

`<h1>Университет Информационных Технологий</h1>`

`<h2>Естественнонаучный факультет</h2>`

`<h3>Расписание занятий</h3>`

Университет Информационных Технологий

Естественнонаучный факультет

Расписание занятий

Всего может быть до шести уровней заголовков (от `<h1>` до `<h6>`)

Горизонтальная черта (разделитель) – блочный элемент

`<p>Университет Информационных Технологий</p><hr />`

`<p>Естественнонаучный факультет</p>`

Университет Информационных Технологий

Естественнонаучный факультет

Гиперссылка

Гиперссылка – строчный элемент

`<p>Пользуйтесь поиском`

`Google`

`– самый распространенный поисковик в мире`

`</p>`

`<p>Итак, давайте поиграем!</p>`

Пользуйтесь поиском [Google](#) - самым распространенным поисковиком в мире!

Будьте аккуратны с вложенностью элементов!

`<p>Это в первом абзаце</p>`

`<p>А это уже во втором!</p>`

Здесь две ошибки: блочный элемент внутри строчного и неправильная вложенность элементов. Тем не менее, HTML-браузер может корректно отобразить эти элементы!

Картинки

Вставка изображений – строчный элемент

```
<p></p>
```



Дополнительные атрибуты

```

```

```

```

Изображение может служить ссылкой так же, как и текст:

```
<a href="servers/server.html">  
    
</a>
```

Списки

Список (нумерованный или ненумерованный) – блочный элемент, Содержащий внутри себя блочные элементы – члены списка

```
<ul>
  <li>Первая строка списка</li>
  <li>Вторая строка списка</li>
  <li>Третья строка списка</li>
</ul>
```

- Первая строка списка
- Вторая строка списка
- Третья строка списка

Атрибуты списка могут задавать вид маркеров (не поддерживается в «строгом» XHTML), например

```
<ul type="square">
  <li>Первая строка списка</li>
</ul>
```

- Первая строка списка

Нумерованные списки

Нумерованный список

```
<ol>  
  <li>Вымыть посуду</li>  
  <li>Постирать носки</li>  
  <li>Купить жене цветы</li>  
</ol>
```

1. Вымыть посуду
2. Постирать носки
3. Купить жене цветы

*Атрибуты списка могут задавать тип и способ нумерации
(не поддерживается в «строгом» XHTML)*

```
<ol start="3" type="a">  
  <li>Сделать домашнее задание</li>  
  <li>Купить билеты на концерт</li>  
</ol>
```

- c. Сделать домашнее задание
- d. Купить билеты на концерт

Вложенные списки

Один список может быть вложен в другой

```
<ol>
  <li>Документы:
    <ul>
      <li>Паспорт</li>
      <li>Билеты</li>
      <li>Гостиница - бронь</li>
    </ul>
  </li>
  <li>Предметы личной гигиены:
    <ul>
      <li>Зубная щетка</li>
      <li>Паста</li>
      <li>Мыло</li>
    </ul>
  </li>
</ol>
```

1. Документы:
 - Паспорт
 - Билеты
 - Гостиница - бронь
2. Предметы личной гигиены:
 - Зубная щетка
 - Паста
 - Мыло

Выделение фрагментов текста

Выделение производится с помощью тегов ``, ``, `<kbd>`, `<code>`

`<p>`Изучая `<code>`HTML`</code>` следует обратить ``особое внимание`` на различие между ``нумерованными`` списками `<kbd>````</kbd>` и ``ненумерованными`` списками `<kbd>````</kbd>`.`</p>`

Изучая HTML следует обратить **особое внимание** на различие между нумерованными списками `` и ненумерованными списками ``.

Фрагменты кода можно выделять с помощью блочного тега `<pre>`

```
<pre>
    <strong>public static void</strong> main(String[] args) {
        System.out.println("Hello, World!");
    }
</pre>
```

Этот фрагмент выводится в точности так же, как он напечатан, включая все пробелы и переводы строк.

Элементы заголовка HTML

`<title>Это моя страница</title>`

Отображается в заголовке страницы в браузере.

`<meta name="description"`

`content="Эта страница содержит полезные сведения"/>`

Описывает содержимое страницы.

`<meta name="keywords" content="мир, труд, май"/>`

Описывает ключевые слова (часто используется поисковыми машинами).

`<meta name="author" content="Александр Кубенский"/>`

Указание на автора.

`<meta name="generator" content="Microsoft FrontPage 8.0"/>`

Указание на программу, которая сгенерировала этот текст.

Элементы заголовка HTML

```
<meta http-equiv="Content-Language" content="ru">
```

Определяет основной язык, на котором написана страница.

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

Описывает формат и кодировку страницы.

```
<meta http-equiv="refresh"
      content="5;http://www.google.com/">
```

Описывает частоту перезагрузки страницы (в секундах) и, возможно, делает “redirection” на другую страницу.

```
<link type="image/jpeg" rel="shortcut icon"
      href="favicon.jpg">
```

Описывает иконку, ассоциирующуюся со страницей (устаревший вариант: поместить в корневой каталог сайта файл с именем favicon.ico).

MIME-типы

Internet media type (MIME – Multipurpose Internet Mail/Media Extension)

<code>application/javascript</code>	Текст на JavaScript (обычно .js)
<code>application/octet-stream</code>	Двоичная программа (обычно .exe)
<code>image/jpeg</code>	Картинка в формате JPEG
<code>image/gif</code>	Картинка в формате GIF
<code>image/vnd.microsoft.icon</code>	Иконка в формате Microsoft
<code>audio/mpeg</code>	Звуковой файл (обычно .mp3)
<code>text/plain</code>	Просто текст (например, .txt)
<code>text/html</code>	Текст на языке HTML
<code>text/css</code>	Каскадная таблица стилей (.css)
<code>video/mp4</code>	Кино в формате MP4

Подробнее см. <http://ru.wikipedia.org/wiki/MIME>

Мнемоники

Специальные символы отображаются в текстах в виде специальных слов (entities). Вот примеры некоторых мнемоник:

< >	< >
™ ©	™ ©
π δ Δ	π δ Δ
И	&1048;
¢ £ ¥	¢ £ ¥
" &	" &

Полный список см.

http://www.w3schools.com/tags/ref_entities.asp