

Лекция 1

Тема: Язык гипертекстовой разметки HTML. Форматирование HTML-документов.

Цель : познакомить с языком гипертекстовой разметки HTML.

Спецификации HTML

Работа по созданию спецификации HTML ведется организацией World Wide Web Consortium (сокращенно — W3C).

В настоящее время существуют спецификации 2.0, 3.0, 3.2 и 4.0.

В спецификации HTML 4.0 ключевой идеей стало отделение описания структуры документа от описания его представления на экране монитора. Разделение структуры и представления документа уменьшает затраты на поддержку широкого спектра платформ, сред и т. п., а также облегчает внесение исправлений в документы. В соответствии с этой идеей следует шире пользоваться методами описания представления документа с помощью таблиц стилей, вместо того, чтобы задавать конкретные данные о форме представления вперемешку с содержанием документа. Для реализации этой идеи в спецификации HTML 4.0 ряд тэгов, используемых для непосредственного задания формы представления HTML-элементов, отменены.

Страница Web должна включать начальные строки кода для указания соответствия стандарту, применяемому к странице.

Структура HTML-документа

HTML-документ представляет собой текстовый файл с расширением *.html или *.htm.

Для удобства используются необязательные дополнительные отступы. Браузеры игнорируют символы конца строки и множественные пробелы в HTML-файлах.

Элементы и тэги

HTML состоит из элементов, которые определяют, каким образом данные должны быть отображены в браузере. Элемент состоит из открывающего и, опционально, закрывающего тэгов:

`<elementName>... some content ...</elementName>`

Как видно из примера, вся информация о форматировании документа сосредоточена в его фрагментах, заключенных между знаками "<" и ">". Такой фрагмент (например, `<html>`) называется меткой (по-английски — tag, читается "тэг").

Тег — часть HTML, представляющая из себя текст, заключенный в угловые скобки `<>`.

В соответствии со стандартами XHTML все ключевые слова записываются символами нижнего регистра. Большинство тегов XHTML кодируются парами из открывающего и закрывающего тегов, называемых контейнерными тегами. Открывающий тег является самым ключевым словом, появляющимся между символами `<` и `>` ; закрывающий тег имеет такой же формат, в котором перед ключевым словом стоит прямая косая черта (`/`). Эта пара контейнерных тегов охватывает данные, к которым применяется форматирование.

Смысл HTML-элемента состоит в применении форматирования к содержимому между начальным и конечным тэгами. Применяемое форматирование зависит от имени элемента.

Не все HTML-элементы имеют открывающий и закрывающий тэг, то есть не все теги являются контейнерными, а кодируются единственным тэгом. Такие пустые тэги кодируются специальным образом — они должны содержать символ наклонной черты (`/`) непосредственно перед закрывающей угловой скобкой.

Тэг `<html>` должен открывать HTML-документ. Аналогично, тэг `</html>` должен завершать HTML-документ. Документ может состоять из двух разделов — раздела заголовка (начинающийся тэгом `<head>`) и раздела содержательной части документа (начинающийся тэгом `<body>`). Для документов, описывающих фреймовые структуры, вместо раздела `body` используется раздел `frameset` (с тэгом `<frameset>`).

Параметры тэгов

Многие тэги, помимо имени, могут содержать или атрибуты (от англ. attribute) или параметры — элементы, дающие дополнительную информацию о том, как браузер должен обработать текущий тэг.

Наборы допустимых параметров индивидуальны для каждого тэга. Общие правила записи параметров заключаются в следующем. После имени тэга могут следовать параметры, которые отделяются друг от друга пробелами. Порядок следования параметров тэга произволен. Многие параметры требуют указания их значений, однако некоторые параметры не имеют значений или могут записываться без них, принимая значения по умолчанию. Если параметр требует значения, то оно указывается после названия параметра через знак равенства. Значение параметра может записываться в кавычках, так и без них. Единственным случаем, когда использование кавычек обязательно, является случай, когда в значении параметра имеются пробелы. В значениях параметров (в отличие от названий тэгов и самих параметров) иногда важен регистр записи.

Приведем пример записи тэга с параметрами:

```
<table border align="left">
```

Здесь для тэга <table> задано два параметра. Первый параметр border указан без значения. Второй параметр align имеет значение " left ".

Обработка ошибочных и неизвестных тэгов

Существуют общие правила интерпретации тэгов браузерами. В отличие от языков программирования, в которых ошибочные операторы приводят к выдаче соответствующих сообщений на этапе компиляции программы и требуют правки, в HTML не принято реагировать на неверную запись тэгов. Неверно записанный тэг или его параметр должен просто игнорироваться браузером. Это общее правило для всех браузеров, под действие которого попадают не только ошибочно записанные тэги, но и тэги, не распознаваемые данной версией браузера. Примером могут служить тэги, предложенные и реализованные для отдельного браузера и неизвестные для другого. Например, тэг-контейнер <noframes>, который служит для предоставления альтернативной информации браузерам, не обеспечивающим поддержку фреймовых структур, такими браузерами не будет распознан.

Заголовок HTML-документа

Раздел документа head определяет его заголовок и не является обязательным тэгом. Задачей заголовка является представление необходимой информации для программы, интерпретирующей документ. Тэги, находящиеся внутри раздела head (кроме названия документа, описываемого с помощью тэга <title>), не отображаются на экране.

Пара меток <head> ... </head> указывает на начало и конец заголовка документа. Помимо наименования документа (см. <title>), в этот раздел может включаться множество служебной информации.

Название документа

Тэг-контейнер <title> является единственным обязательным тэгом заголовка и служит для того, чтобы дать документу название. Оно обычно показывается в заголовке окна браузера. Все, что находится между метками <title> и </title>, интерпретируется браузером как название документа. Рекомендуется название не длиннее 64 символов.

По умолчанию текст, содержащийся в названии документа, используется при создании закладки (bookmark) для документа. Поэтому, для большей информативности, следует избегать безликих названий (HomePage, Index и т. д.). Название документа должно кратко характеризовать его содержание.

Связи между HTML-документами

Часто HTML-документы связаны между собой, то есть имеют ссылки друг на друга. Ссылки могут быть как абсолютные, так и относительные.

Абсолютные ссылки могут быть громоздкими и переставать работать, если перемещен младший по иерархии документ. Относительные ссылки легче вводить и обновлять, но и эта связь обрывается, если перемещен старший по иерархии документ. Оба вида связей могут нарушиться при переносе документа с одного компьютера на другой.

Для решения этой проблемы используются тэги `<base>` и `<link>`, которые включаются в заголовок для того, чтобы связь между документами не нарушалась.

Тэг `<base>` служит для указания полного базового URL-адреса документа. С его помощью относительная ссылка продолжает работать, если документ переносится в другой каталог или даже на другой компьютер.

Тэг `<base>` имеет один обязательный параметр `href`, после которого указывается полный URL-адрес документа. Пример:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
<title> Указание базового адреса </title>
<base href="//www.volsu.ru/fitt"/>
</head>
<body>

</body>
</html>
```

Тэг `<link>` указывает на связь документа, содержащего данный тэг и другого документа или объекта. Он состоит из URL-адреса и параметров, конкретизирующих отношения документов. Заголовок документа может содержать любое количество тэгов `<link>`. В табл. 2.1 описаны параметры тэга `link`.

Таблица 2.1. Параметры тэга `link`

П араметр	Назначение
<code>href</code>	Указывает на URL-адрес другого документа
<code>rel</code>	Определяет отношение между текущим и другим документом
<code>rev</code>	Определяет отношение между другим документом и текущим (отношение, обратное REL)
<code>type</code>	Указывает тип и параметры присоединенной таблицы стилей

Пример:

```
<link rel="contents" href = "../toc.html"/>
<link href="mailto:info@fitt.volsu.ru" rev="made"/>
```

Первая строка указывает на связь с файлом оглавления документа (`toc.html`— `table of contents`) с прямым отношением `contents`. Вторая строка описывает связь с URL-адресом автора документа (с обратным отношением `made`).

Между документами может существовать множество различных отношений. Примеры других значений параметра `rel`: `bookmark`, `copyright`, `glossary`, `help`, `home`, `index`, `toe`, `next`, `previous`. Параметр `rev` может также принимать значения: `author`, `editor`, `publisher`, `owner`.

Тэг `meta`

В раздел заголовка может быть добавлен еще один тэг `<meta>`, позволяющий авторам документа определять информацию, не имеющую отношения к HTML. Эта информация используется браузером для действий, которые не предусмотрены текущей спецификацией HTML.

Пример:

```
<meta http-equiv="refresh" content="60"/>
```

Этот тэг инструктирует браузер перезагружать страницу каждые 60 секунд, что может быть полезно, если данные на странице часто обновляются.

Кроме того, в качестве примера можно привести указание ключевых слов, используемых поисковыми системами. Этот способ позволяет включать в индекс документа дополнительные слова, которые могут явно не входить в его содержание. Для этого в тэге `<meta>` в качестве значения параметра `name` указывается имя некоторого свойства.

Спецификация HTML не определяет каких-либо конкретных имен свойств, записываемых в тэге `<meta>`. Однако есть несколько часто применяемых свойств, например, `description`, `keywords`, `author`, `robots` и др.:

```
<meta name="description" content="Технологии создания интернет-узлов"/>
```

```
<meta name="keywords" content="тэг, гипертекст, HTML, браузер"/>
```

Еще одно назначение тэга `<meta>` — это указание кодировки текста. Например, для текста на русском языке в кодировке Windows используется следующий тэг:

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

Тело документа

Пара меток `<body> ... </body>` указывает на начало и конец тела HTML-документа, которое, собственно, и определяет содержание документа.

Все тэги, которые допустимо использовать в разделе `<body>` документа HTML, могут иметь параметры `class`, `id`, `lang`, `language`, `style` и `title`. Использование этих параметров件件件, прежде всего, при стилевом оформлении документов.

В современном HTML, помимо тэгов языка и их содержимого, в исходном HTML-коде также записываются коды сценариев (JavaScript или VBScript).

Форматирование текста

Тэги уровня блока и последовательные тэги

Некоторые HTML-тэги, которые могут появляться в разделе `body`, называют тэгами уровня блока (block level), в то время как другие последовательными (inline) тэгами или, называя по-другому, тэгами уровня текста (text level), хотя такое разделение тэгов по уровням в известной степени условно.

Различие уровней HTML-тэгов заключается в следующем: тэги уровня блока могут содержать последовательные тэги и другие тэги уровня блока, тогда как последовательные тэги содержат только данные и другие последовательные тэги. Блочные тэги описывают более крупные структуры документов, по сравнению с последовательными тэгами.

По умолчанию тэги этих видов размещаются в тексте описания различным образом: тэги уровня блока начинаются с новой строки, в то время как последовательные — нет.

HTML допускает два подхода к шрифтовому выделению фрагментов текста. С одной стороны, можно прямо указать, что шрифт на некотором участке текста должен быть жирным или наклонным, то есть изменить физический стиль текста. С другой стороны, можно пометить некоторый фрагмент текста как имеющий некоторый отличный от нормального логический стиль, оставив интерпретацию этого стиля браузеру.

Для форматирования текста HTML-документов предусмотрен ряд тэгов, которые можно условно разделить на тэги логического и физического форматирования.

Физические стили

Под физическом стилем понимается прямое указание браузеру на модификацию текущего шрифта. Тэги физического форматирования определяют формат отображения указанного в них фрагмента текста в окне браузера.

Все, что находится между метками `` и ``, будет написано жирным шрифтом. В большинстве случаев рекомендуется вместо тэга `` использовать тэг логического форматирования ``.

Это ``полужирный`` шрифт

Текст между тэгами `<i>` и `</i>` будет написан наклонным шрифтом. Для большинства случаев вместо этого тэга рекомендуется использовать тэги ``, `<dfn>`, `<var>` или `<cite>`, поскольку последние лучше отражают назначение выделяемого текста.

Текст, размещенный между тэгами `<tt>` и `</tt>`, будет написан шрифтом, имитирующим пишущую машинку, то есть имеющим фиксированную ширину символа. Для большинства случаев вместо этого тэга лучше использовать тэги `<code>`, `<samp>` или `<kbd>`.

Тэг `<u>` отображает текст подчеркнутым. Отмененный тэг. Вместо него рекомендуется использовать тэги `` или `<cite>`.

Параметры шрифта

Тэг `... `, позволяет явно указать параметры шрифта для фрагмента страницы. Этот тэг имеет атрибуты `face`, который определяет имя шрифта, `size` – размер шрифта и `color` – цвет шрифта. Можно указать как один, так и несколько названий шрифтов, разделяя их запятыми. Список шрифтов просматривается слева направо. Если на компьютере пользователя нет шрифта, указанного в списке первым, то делается попытка найти следующий шрифт.

```
<font face="Verdana", "Arial", "Helvetica">
```

Этот текст написан заданным шрифтом

```
</font>
```

Размер шрифта указывается в условных единицах от 1 до 7. Конкретный размер шрифта зависит от используемой программы просмотра. Принято считать, что размер обычного шрифта соответствует значению 3.

```
<font size="1">Размер 1</font><br />
```

```
<font size="3">Размер 3</font><br />
```

```
<font size="5">Размер 5</font><br />
```

Цвет шрифта может задаваться с помощью стандартных имен или в формате `#RRGGBB`.

Пример:

```
<font color="green">Зеленый</font> <br />
```

```
<font color="#FF0000">Красный</font> <br />
```

Тэг `<basefont>` используется для указания размера, типа и цвета шрифта, используемого в документе по умолчанию. Эти значения обязательны для всего документа, однако могут в нужных местах переопределяться с помощью тэга ``. После закрывающего тэга `` действие тэга `<basefont>` восстанавливается. Значения параметров шрифтов, используемых по умолчанию, могут неоднократно переопределяться в документе, т. е. тэг `<basefont>` может появляться в документе любое количество раз. Для тэга `<basefont>` не существует закрывающего тэга. В качестве параметров могут использоваться те же параметры, что и для тэга ``.

```
<basefont color = "green">
```

Назначение параметров шрифта непосредственно в тексте документа нарушает основную идею разделения содержательной части документа и описания формы представления документа. Поэтому в спецификации HTML 4.0 данный тэг, а также тэг `<basefont>` отнесены к отмененным. Их дальнейшее применение не рекомендуется.

Разделение текста на абзацы

Любой текст имеет определенную структуру, элементами которой являются заголовки, подзаголовки, таблицы, абзацы и др.

При создании документов с помощью текстовых редакторов разбиение на абзацы выполняется вводом символа перевода строки. Большинство редакторов реализует это при нажатии клавиши <Enter>. В HTML-документах символы перевода строки не приводят к образованию нового абзаца.

Тэги вида <h*i*> (где *i* — цифра от 1 до 6) описывают заголовки шести различных уровней. Заголовок первого уровня — самый крупный, шестого уровня самый мелкий. Тэги заголовка являются элементами уровня блока, поэтому с помощью них нельзя размечать отдельные слова текста для увеличения их размера.

Пара тэгов <p>и </p>описывает абзац. Все, что заключено между ними, воспринимается как один абзац. Закрывающий тэг в HTML 4.0 </p>не обязателен, однако в XHTML его следует использовать.

Тэги <h*i*>и <p>могут содержать дополнительный атрибут align (табл. 2.2), например:

```
<h1 align="center">
```

Выравнивание заголовка по центру

```
</h1>
```

```
<p align="right">
```

Абзац с выравниванием по правому краю

```
</p>
```

Таблица 2.2. Значения параметра align

Значение параметра align	Действие
left	Выравнивание текста по левой границе
center	Выравнивание по центру
right	Выравнивание по правой границе
justify	Выравнивание по ширине

Использование атрибута align в XHTML также не рекомендуется.

При отображении текстовых документов в браузере место переноса строки в пределах абзаца определяется автоматически в зависимости от размера шрифтов и размера окна просмотра. Перенос строки может осуществляться только по символам-разделителям слов (например, пробелам). Иногда в документах требуется задать принудительный перевод строки, реализующийся независимо от параметров настроек браузера. Тэг
используется, если необходимо перейти на новую строку, не прерывая абзаца. В отличие от тега <p> ... </p>этот тег не приводит к вертикальному отступу между строками.

Другим методом разделения документа на части является проведение горизонтальных линий. Они визуально подчеркивают законченность той или иной области страницы. Тэг <hr/>описывает горизонтальную линию между параграфами. Тэг может дополнительно включать ряд атрибутов (табл. 2.3).

Таблица 2.3. Атрибуты тэга <hr>

Параметр <hr/>	Назначение
align	Выравнивает по краю или центру; имеет значения left, center, right
width	Устанавливает длину линии в пикселах или процентах от ширины страницы
size	Устанавливает толщину линии в пикселах

noshade	Отменяет рельефность линии
color	Указывает цвет линии. Используется формат RGB или стандартное имя

Все перечисленные атрибуты тэга `<hr/>` в XHTML являются устаревшими. Пример:
`<hr align="center" width="50%" noshade />`

Бывают случаи, когда в HTML-документ необходимо включить текст, уже имеющий форматирование, выполненное традиционным способом при помощи символов перевода строки, необходимого количества пробелов, символов табуляции и т. д. Для решения таких задач предусмотрен специальный тэг-контейнер `<pre>`, определяющий предварительно форматированный (преформатированный) текст. Текст, размеченный тэгом `<pre>`, будет отображаться в таком виде, как он выглядит в обычном текстовом редакторе. Для отображения будет использоваться моноширинный шрифт. Пример:

```
<pre>
один
два
три
</pre>
```

Логические стили

При использовании логических стилей автор документа не может знать заранее, что увидит на экране читатель. Разные браузеры интерпретируют одни и те же тэги логических стилей по-разному. Некоторые браузеры игнорируют некоторые тэги вообще и показывают обычный текст вместо выделенного логическим стилем.

Речь идет о структурной разметке, которая не влияет на конкретное экранное представление фрагмента браузером. Поэтому такая разметка и называется логической. Фрагменты с логическим форматированием браузеры отображают на экране определенным образом, заданным по умолчанию. Вид отображения никак не связан со структурным типом фрагмента (т. е. именем тэга логического форматирования), но может быть легко переопределен.

Тэги логического форматирования обозначают структурные типы текстовых фрагментов, такие, например, как программный код (тэг `<code>`), цитата (тэг `<cite>`), аббревиатура (тэг `<abbr>`) и т. д.

Самые распространенные логические стили:

` ... ` - от английского *emphasis* — акцент. Используется для выделения важных фрагментов текста. Браузеры обычно отображают такой текст курсивом.

Пример выделения `` отдельных слов `` текста

` ... ` - от английского *strong emphasis* — сильный акцент. Используется для выделения важных фрагментов текста. Браузеры обычно отображают такой текст полужирным шрифтом.

Этот фрагмент `` очень важен ``

`<code> ... </code>` - рекомендуется использовать для фрагментов исходных текстов. Отмечает текст как небольшой фрагмент программного кода. Как правило, отображается моноширинным шрифтом. Пример:

```
<code>Response.Write("Hello, World!")</code>
```

`<samp> ... </samp>` - от английского *sample* — образец. Отмечает текст как образец. Рекомендуется использовать для демонстрации образцов сообщений, выводимых на экран программами. Для выделения используется моноширинный шрифт.

Результат работы программы: `<samp>Hello, World!</samp>`

`<kbd> ... </kbd>` - от английского *keyboard* — клавиатура. Рекомендуется использовать для указания того, что нужно ввести с клавиатуры. Обычно отображается моноширинным шрифтом.

Введите с клавиатуры:

```
<kbd>Hello, World!</kbd>
```

`<var> ... </var>`От английского *variable* — переменная. Рекомендуется использовать для написания имен переменных. Обычно такой текст отображается курсивом.

Объявите переменную `<var>a</var>`

Использование логического форматирования предпочтительнее, так как с применением принципа отделения структуры документа от его представления можно гибко управлять визуализацией документа, используя, например, таблицы стилей. Эта технология позволяет, в частности, создавать представления одного и того же документа для полнофункционального РС-браузера, браузера карманного компьютера, сотового телефона и т.п., а также позволяет пользователю самостоятельно выбирать схему цветового и шрифтового оформления документа. Кроме того, использование логического форматирования позволяет вставлять документы с одного сайта в другой, применяя к ним новое оформление.

Тэг `<div>`

Тэг-контейнер `<div>` является элементом уровня блока, служащим для выделения фрагмента документа. Целью этого выделения является управление параметрами данного фрагмента, которое обычно выполняется с помощью назначения стилей.

```
<div style="color: green">
```

(Фрагмент документа)

```
</div>
```

В этом примере фрагмент HTML-документа обрамляется тэгами `<div>` и `</div>` для задания некоторых его свойств. В данном случае все текстовые элементы выделенного фрагмента будут отображаться зеленым (green) цветом. Аналогом тэга `<div>` уровня текста является элемент ``.

Непосредственное назначение стилевых свойств отдельного фрагмента так, как это сделано в примере, использовать не желательно в соответствии с концепцией разделения структуры документа и его представления. Следует использовать таблицы стилей.

Комментарии

В HTML-документ можно включать комментарии, которые не будут видны читателю. Они могут состоять из произвольного числа строк и должны начинаться тэгом `<!--` и заканчиваться тэгом `-->`. Все, что заключено внутри этих тэгов, при просмотре страницы не будет отображаться на экране.

Комментарии в HTML применяются также для того, чтобы скрыть от браузера скрипты в случае, если он не в состоянии распознать их.

Специальные символы

Некоторые специальные символы не входят в базовую часть таблицы кодов ASCII. К ним относятся буквы алфавитов части европейских языков, математические и некоторые другие символы. В таких случаях следует включать нужные символы в HTML-документ при помощи специальных кодов. Эти коды состоят из символа амперсанда (&) и следующим за ним именем символа или его десятичным или шестнадцатеричным значением. Заканчиваться специальный символ должен знаком "точка с запятой".

В спецификации HTML приводятся целые таблицы со специальными символами и их значениями. На сегодняшний день браузеры распознают лишь небольшое количество специальных символов (табл. 2.4).

Таблица 2.4. Спецсимволы HTML

Запись специального символа

Назначение

<

<

>	>
 	Неразрывный пробел
©	©
&	&
"	"

Все символами могут быть также заданы своими кодами. Например, символ неразрывного пробела имеет код 160. Он может записываться в десятичном виде как ` `.