

1. Лекция: Введение в HTML

В этой лекции рассказывается о языке гипертекстовой разметки HTML, вводится понятие "тег" и рассматривается пример простейшей web-страницы, подробно рассматриваются теги и приводятся примеры их параметров. Рассматривается структура web-страницы. Рассматриваются теги параграфа, заголовка, отступа и способы организации комментария.

Коротко о HTML

HTML (Hyper Text Markup Language) означает язык разметки гипертекста. Этот язык был разработан Тимом Бернерсом-Ли в рамках создания проекта распределенной гипертекстовой системы, которую он назвал World Wide Web (WWW) или Всемирная паутина. HTML предназначен для написания гипертекстовых документов, публикуемых в World Wide Web. Документ на языке HTML может включать следующие компоненты:

- стилизованный и форматированный текст,
- команды включения графических и звуковых файлов,
- гиперсвязи с различными ресурсами Internet.
- скрипты на языке JavaScript и VBScript.
- различные объекты, например Flash-анимацию

Документы HTML являются обычными текстовыми файлами, содержащими специальные теги (или управляющие элементы) разметки. Теги разметки указывают браузеру Web (программе пользователя для отображения web-страниц, например, Internet Explorer, Mozilla, Netscape или Opera), как надо вывести страницу.

Файлы HTML обычно имеют расширения htm или html. Их можно создавать при помощи любого текстового редактора.

Язык HTML является подмножеством мощного языка SGML (Standard Generalized Markup Language), который широко используется в издательской деятельности. Основной выигрыш от использования этих языков состоит в переносимости текста между разными издательскими системами. Эта же особенность сохраняется и в HTML. Так, читая документ, пользователи могут устанавливать способы выделения текста, гарнитуру и размер шрифтов по своему вкусу; они могут отменить просмотр рисунков.

В документе HTML можно выделить **два основных блока: головная часть и тело документа**. Содержимое головной части не выводится на экран пользователя, за исключением заголовка, в ней, как правило, указывают ключевые слова, авторов и другую служебную информацию, а также подключают внешние таблицы стилей и скрипты. В теле документа размещают ту информацию которая будет выведена пользователю.

Мой первый сайт

В разных операционных системах имеются различные редакторы, которые можно использовать для создания документов HTML. Если вы используете:

- Microsoft Windows запустите редактор Notepad;
- Mac OS запустите редактор SimpleText;
- OSX запустите редактор TextEdit (обязательно измените настройки "Rich Text" или "Расширенный текст" на "Plain text" или "Простой текст" и отметьте пункт "Ignore rich text commands in HTML files" или "Игнорировать команды расширенного текстового формата в файлах HTML").

Файлы HTML можно создавать и в редакторе Microsoft Word, в котором имеется возможность сохранить документ как Web-страницу (в меню "Файл"), однако использовать эту возможность не рекомендуется. Во-первых, потому что HTML-код, генерируемый MS Word не оптимален и содержит множество ненужных элементов разметки, и, во-вторых, автоматическая генерация кода не будет способствовать изучению и правильному пониманию HTML.

Имеется также большое количество специализированных редакторов для создания файлов HTML, таких как FrontPage, Macromedia Dreamweaver или Adobe Web Bundle, которые обладают возможностью WYSIWYG (What You See Is What You Get - что видишь, то и получишь). С их помощью можно легко создавать документы HTML, при помощи кнопок и элементов меню, а не писать самому теги разметки. Однако, как уже отмечалось выше, тем, кто хочет стать технически грамотным разработчиком Web, настоятельно рекомендуется использовать простой текстовый редактор для начального изучения HTML.

Задание 1. Напечатайте следующий текст:

```
<html>
<head>
<title>Это заголовок страницы</title>
</head>
<body>
<h1>Здравствуй!</h1>
<p>Это моя первая страница HTML. <b>Этот текст выводится жирным
шрифтом.</b></p>
</body>
</html>
```

Сохраните файл как "page1.htm".

При сохранении файла HTML можно использовать расширение .htm или .html. Расширение .htm было принято для старых версий операционных систем, которые допускали трехбуквенное расширение для файлов. В настоящее время

практически все операционные системы не имеют подобного ограничения и можно использовать расширение .html.

Теперь посмотрите как браузер отобразит вашу первую страницу. Запустите браузер Интернет. Выберите "Open" или "Open Page" ("Открыть" или "Открыть страницу") в меню File (Файл) браузера. Появится диалоговое окно. Выберите "Browse" или "Choose File" ("Просмотр" или "Выбрать файл") и найдите только что созданный файл HTML - "page1.htm" - выберите его и щелкните на кнопке "Open" ("Открыть"). В диалоговом окне должен появиться адрес, например "C:\MyDocuments\page1.htm". Щелкните на кнопке ОК, и браузер выведет на экран вашу страницу.

Разбор примера

Ваш первый HTML-документ начинается с тега <html>, который сообщает браузеру о начале документа HTML и заканчивается тегом </html>, который информирует браузер о достижении конца документа HTML.

Текст между тегами <head> и </head> является информацией заголовка документа. Эта информация не выводится в окне браузера.

Текст "Это заголовок страницы" между тегами <title> и </title> является заголовком документа. Этот заголовок выводится в строке заголовка окна браузера.

Текст между тегами <body> и </body> является текстом, который будет выведен в окне браузера. Текст "Здравствуйте!" между тегами <h1> и </h1> будет отображен стилем заголовка, обычно жирным шрифтом большего размера.

Тег <p> означает, что начинается новый параграф, тег </p> означает конец параграфа.

Текст "Этот текст выводится жирным шрифтом." между тегами и будет выведен жирным шрифтом.

Элементы HTML

Страницы Всемирной паутины отображаются при помощи документов HTML, которые являются обычными текстовыми файлами, содержащими специальные теги разметки. Теги разметки определяют элементы HTML, из которых состоят документы HTML.

Коротко о тегах

Теги HTML используются для выделения элементов HTML. Обычно теги HTML используются парами и заключены между двумя символами угловых скобок < (начальный тег)> и <(конечный тег)>. Текст между начальным и конечным тегами является содержимым элемента. Некоторые теги не имеют конечного,

например, тег принудительного переноса строки `
`, для таких тегов рекомендуется использовать следующее написание `
`.

Регистр символов для отображения тегов не важен, например, `<p>` и `<P>` означает одно и то же. Однако в этом курсе используется нижний регистр для написания тегов. Это связано с тем, что консорциум WWW (W3C), который занимается стандартизацией спецификации HTML, рекомендует использовать теги в нижнем регистре, поскольку в следующем поколении стандартов будет именно такое требование.

Коротко об элементах HTML

Рассмотрим тот же пример документа HTML:

```
<html>
<head>
<title>Это заголовок страницы</title>
</head>
<body>
<h1>Здравствуйте!</h1>
<p>Это моя первая страница HTML. <b>Этот текст выводится жирным
шрифтом.</b></p>
</body>
</html>
```

Элементом HTML является:

```
<h1>Здравствуйте!</h1>
```

Этот элемент начинается с тега `<h1>`, имеет содержимое "Здравствуйте!" и заканчивается тегом `</h1>`.

Также элементом HTML, является:

```
<p>Это моя первая страница HTML. <b>Этот текст выводится жирным
шрифтом.</b></p>
```

Этот элемент, начинается с начального тега `<p>`, заканчивается конечным тегом `</p>` и означает, что содержимое элемента "Это моя первая страница HTML. `` Этот текст выводится жирным шрифтом.``" является отдельным параграфом. При этом внутри этого элемента находится другой элемент:

```
<b>Этот текст выводится жирным шрифтом.</b>
```

Этот элемент HTML начинается с начального тега: `` Содержимым элемента HTML является: Этот текст выводится жирным шрифтом. Этот элемент HTML заканчивается конечным тегом ``. Назначение тега `` состоит в определении элемента HTML, который должен выводиться жирным шрифтом.

Все описанные элементы HTML содержатся в элементе:

```
<body>
<h1>Здравствуйте!</h1>
<p>Это моя первая страница HTML. <b>Этот текст выводится жирным
шрифтом.</b></p>
</body>
```

Этот элемент HTML начинается с начального тега `<body>`, и заканчивается конечным тегом `</body>`. Назначение тега `<body>` состоит в определении элемента HTML, который содержит основную часть (или тело) документа HTML.

Атрибуты тегов

Теги могут иметь атрибуты, которые предоставляют дополнительную информацию об элементах HTML. Атрибуты всегда используются в виде пары "имя/значение". Общий формат задания атрибутов имеет вид:

```
<имя_тега имя_атрибута="значение">
```

Например, тег:

```
<body bgcolor="red">
```

означает, что цвет фона страницы должен быть красным.

А тег:

```
<p align="center">
```

означает, что параграф необходимо выравнивать по центру страницы отображения браузера.

Атрибуты всегда помещаются в начальном теге элемента HTML. Значения атрибутов всегда должны заключаться в кавычки. Наиболее широко используются двойные кавычки, но одиночные кавычки также допустимы.

В некоторых редких ситуациях, когда, например, значение атрибута само содержит кавычки, необходимо использовать одиночные кавычки:

```
<html>
<body>
<abbr title='проект "Майкопский Государственный Гуманитарно-Технический
Колледж" '>МГГТК АГУ</abbr>
</body>
</html>
```

Кроме атрибутов записываемых вышеописанным способом, для некоторых элементов определены специальные флаги, которые просто указываются как <тег имя_флага>.

```
<html>
<body>
<p>Попробуйте исправить текст в этих полях ввода</p>
<form action="index.php" method="GET">
<input type="text" name="blocked" value="Пример поля ввода у которого указан
флаг readonly"
  readonly size="100"><br/><br/>
<input type="text" name="unblocked" value="Обычное поле ввода у которого нет
флагов" size="100">
</form>
</body>
</html>
```

Основные теги HTML

Параграфы.

Прежде чем изучать теги форматирования HTML посмотрим как введенный текст отобразится, если не будут применены никакие теги кроме тегов <html> и <body>. Следующий пример демонстрирует такой документ HTML

```
<html>
<body>
Этот текст будет показан в окне браузера.
</body>
</html>
```

Этот простой пример документа HTML, который содержит минимальное количество тегов HTML и демонстрирует, как текст внутри элемента body отображается в браузере.

Если ввести большой объем текста таким способом, то читать его будет очень не удобно. Логичнее разбить его на параграфы, как в книге, которые повышают читабельность текста, и кроме того выделяют смысловые блоки.

Следующий пример показывает как отображаются параграфы

```
<html>
<body>
<p>Это параграф 1.</p>
<p>Это параграф 2.</p>
<p>Это параграф 3.</p>
</body>
```

</html>

Этот пример демонстрирует, как в браузере выводится текст внутри элементов параграфа. Можно видеть, что по умолчанию текст каждого параграфа выводится в виде отдельного блока. Каждый из таких блоков отделяется от предыдущих и последующих блоков страницы пустой строкой. Однако отображение параграфа браузером может быть легко изменено посредством таблицы стилей.

Можно заметить, что параграфы можно записывать без закрывающего тега </p>, однако лучше этого не делать, тем более, что в следующей версии HTML не позволит пропустить ни один не закрывающий тег.

В разных браузерах на разных мониторах с разным разрешением страница будет отображаться по-разному, поэтому не стоит форматировать при помощи добавления пустых строк и пробелов. Любое число пробелов заменяется одним.

Использование пустых параграфов <p> для вставки пустых строк является плохим стилем, вместо этого используйте тег
.

Заголовки.

Заголовки определяются с помощью тегов от <h1> до <h6>. <h1> определяет заголовок самого большого размера, а <h6> определяет заголовок самого маленького размера.

```
<h1>Это заголовок первого уровня</h1>
<h2>Это заголовок второго уровня</h2>
<h3>Это заголовок третьего уровня</h3>
<h4>Это заголовок четвертого уровня</h4>
<h5>Это заголовок пятого уровня</h5>
<h6>Это заголовок шестого уровня</h6>
```

Заголовки автоматически отделяются дополнительными промежутками от остальных элементов документа.

Переносы строк.

Для переноса внутри параграфа используется тег
, который выполняет принудительный перенос строки.

```
<html>
<body>
<p>Это <br>пара<br>граф с переносами строк</p>
</body>
</html>
```

Тег
 не имеет закрывающего тега. Поэтому для совместимости с будущими версиями стандарта рекомендуется следующее написание тега

Горизонтальная линейка

Разделять различные элементы можно при помощи горизонтальной линейки, для этого используйте тег `<hr>`:

```
<html>
<body>
<p>Это параграф отобразится сверху горизонтальной полосы.</p>
<hr>
<p>Это параграф отобразится снизу горизонтальной полосы.</p>
</body>
</html>
```

Тег `<hr>` не имеет закрывающего тега. Поэтому для совместимости с будущими версиями стандарта рекомендуется следующее написание тега `<hr/>`. Для этого тега определен ряд атрибутов, но они являются устаревшими. И хотя их применение возможно, но консорциум W3C их использовать не рекомендует. Вместо них следует использовать таблицы стилей.

Комментарии в HTML

Тег комментария используется для вставки комментариев в исходный код HTML. Комментарии будут проигнорированы браузером. Комментарии можно использовать для пояснения кода, что может помочь при редактировании исходного кода в будущем.

```
<!-- Это комментарий -->
```

Вот пример:

```
<html>
<body>
Этот текст будет показан в окне браузера.
<!-- Этот текст не будет показан, это комментарий. -->
</body>
</html>
```

Дополнительные параграфы

Этот пример демонстрирует некоторые особенности поведения по умолчанию элементов параграфа.

```
<html>
<body>

<p>
Этот параграф
содержит много строк
в исходном коде,
```


но браузер
это игнорирует.
</p>

<p>
Этот параграф
содержит много пробелов
в исходном коде,
но браузер
это игнорирует.
</p>

</body>
</html>

Перенос строк

Этот пример демонстрирует использование переноса строк в документе HTML.

<html>
<body>

<p>
Чтобы выполнить перенос
строк
в
параграфе,
используйте тег br.
</p>

</body>
</html>

Этот пример демонстрирует некоторые проблемы с форматированием HTML.
Попробуем сформатировать стихи:

<html>
<body>

<p>
В лесу родилась елочка.
В лесу она росла.
Зимой и летом стройная,
Зеленая была.
</p>

<p>Обратите внимание, что браузер просто проигнорировал использованное
форматирование!</p>

</body>
</html>

Заголовки

Этот пример демонстрирует теги, которые выводят заголовки в документе HTML.

```
<html>
<body>

<h1>Это заголовок уровня 1</h1>
<h2>Это заголовок уровня 2</h2>
<h3>Это заголовок уровня 3</h3>
<h4>Это заголовок уровня 4</h4>
<h5>Это заголовок уровня 5</h5>
<h6>Это заголовок уровня 6</h6>

<h1 align="center">Это заголовок 1, Он выровнен по центру страницы.</h1>

<p>Используйте теги заголовков только для заголовков.
Не используйте их просто для того, чтобы выделить что-то жирным шрифтом.
Используйте для этого другие теги.</p>

</body>
</html>
```

Горизонтальная линейка

Этот пример демонстрирует, как использовать горизонтальную линейку.

```
<html>
<body>
<p>Тег <hr> определяет горизонтальную линейку :</p>
<hr>
<p>Это параграф</p>
<hr>
<p>Это параграф</p>
</body>
</html>
```

Комментарий

Этот пример демонстрирует, как использовать комментарий в исходном коде HTML.

```
<html>
<body>

<!--Этот комментарий выводиться не будет-->
<p>Это обычный параграф.</p>
```

```
</body>
</html>
```

Фоновый цвет

Этот пример демонстрирует использование цветного фона на странице HTML. При выборе фона всегда проверяйте, чтобы текст был хорошо читаем!

```
<html>
<body bgcolor="yellow">
<h2>Смотри: Цветной фон!</h2>
</body>
</html>
```

Пример выполнения данного HTML-кода

Таблица основных тегов HTML

Тег	Описание
<html>	Определяет документ HTML
<body>	Определяет основную часть или тело документа
<h1> -- <h6>	Определяет заголовки с 1 по 6
<p>	Определяет параграф
 	Вставляет единичный перенос строки
<hr>	Определяет горизонтальную линейку
<!-->	Определяет комментарий