

УВАЖАЕМЫЕ СТУДЕНТЫ! Законспектируйте в своей рабочей тетради по дисциплине приведенную лекцию (объемом 4-5 страницы), ответьте письменно на контрольные вопросы.

Результаты работы, фотоотчет, предоставить преподавателю на e-mail: igor-gricenko-95@mail.ru **в течении ТРЕХ дней.**

При возникновении вопросов по приведенному материалу обращаться по следующему номеру телефона: (072)132-63-42

ВНИМАНИЕ!!! При отправке работы, не забывайте указывать **ФИО студента, наименование дисциплины, дата проведения занятия (по расписанию).**

Лекция № 2

Тема «Язык разметки гипертекста HTML»

План лекции:

1. Язык разметки гипертекста HTML (понятия тега, разметки, верстки, основные теги разметки, табличная верстка, секционная верстка)

I. Основные сведения об HTML.

За последние годы разработки для Интернета эволюционировали от статических страниц до динамических информационных систем. Некоторое время назад создание современных Web-страниц не требовало практически ничего, кроме совершенного владения языком разметки гипертекста (Hypertext Markup Language, HTML).

HTML представляет собой простой язык обработки текстов; на этом языке при помощи набора тегов (*tags*) создается документ, который можно просматривать специальной программой просмотра Web (*browser*).

HTML — не язык программирования в том смысле, как C++ или Visual Basic; он больше напоминает средства форматирования документов с использованием управляющих последовательностей. Кодирование на HTML часто сравнивают с созданием документа в формате Microsoft Word путем

набивки кодов форматирования прямо в Notepad. Очевидно, что функциональность этого крайне мала.

Под **гипертекстовым документом** понимают документ, содержащий ссылки на другой документ. Реализовано все это через **протокол передачи гипертекста HTTP** (Hyper Text Transfer Protocol).

Информация в документах Web может быть найдена по ключевым словам. Это означает, что каждый обозреватель Web содержит определенные ссылки, через которые образуются гиперсвязи, позволяющие миллионам пользователей Internet вести поиск информации по всему миру.

Гипертекстовые документы создаются на базе **языка HTML** (**Hyper Text Markup Language**). Этот язык весьма прост, управляющие коды его, которые, собственно, и компилируются обозревателем для отображения на экране, состоят из текста ASCII. Ссылки, списки, заголовки, картинки и формы называются **элементами языка HTML**.

В настоящее время существует масса редакторов Web-страниц, которые не требуют от вас знаний основ HTML. Но для того чтобы уметь профессионально подготавливать гипертекстовые документы, вы должны знать их внутреннее строение, то есть код документа HTML.

HTML позволяет вам формировать различную гипертекстовую информацию на основе структурированных документов.

Обозреватель определяет сформированные ссылки и, через протокол передачи гипертекста HTTP, открывает доступ к вашему документу другим пользователям Internet. Разумеется, для успешной реализации всего этого необходим софт, полностью совместимый с WWW и поддерживающий HTML.

II. Описание HTML

HTML-документ - это *обычный текстовой файл*. Используя любой Web-обозреватель, вы можете просмотреть результат вашей работы, просто загрузив в него созданный на основе синтаксиса HTML текстовой файл.

Гипертекстовый язык предоставляет только информацию для чтения. Это означает, что редактировать Web-страницы может лишь тот, кто их создал, а не простой пользователь Сети.

Самый главный элемент гипертекстового языка - это ссылки. В мировой паутине вы просто нажимаете на ссылку и мгновенно оказываетесь в другой точке земного шара на выбранной вами страничке.

Тэги.

Тэг - оформленная единица HTML-кода.

Тег HTML состоит из следующих друг за другом в определенном порядке элементов:

- левой угловой скобки < (такого же, как "меньше чем" символа)
- необязательного слэша /, который означает, что тег является *конечным тегом, закрывающим некоторую структуру*. Таким образом в этом контексте Вы можете читать символ /, как *конец...*

- имени тега, например TITLE или PRE
- необязательных, если даже тег может иметь их, **атрибутов**. Тег может быть без атрибутов или сопровождаться одним или несколькими атрибутами, например: ALIGN=CENTER

- правой угловой скобки > (такой же, как символа "больше чем").

Большинство тегов имеют **открывающий элемент** <> и **закрывающий** </>. Между ними и находятся **коды**, которые распознает Web-обозреватель

В таких случаях два тега и часть документа, отделенная ими, образуют блок, называемый **HTML элементом**. Некоторые теги, например <HR>, являются элементами HTML сами по себе, и для них соответствующий конечный тег неверен.

Для каждого тега определяется множество возможных **атрибутов**. Большинство тегов допускает один или несколько атрибутов, однако атрибутов может и совсем не быть. **Спецификация атрибута** состоит из расположенных в следующем порядке:

- имени атрибута, например WIDTH
- знак равенства (=)
- значения атрибута, которое задается строкой символов, например, "80".

Всегда полезно **заключить значение атрибута в кавычки**, используя либо одинарные ('80'), либо двойные кавычки ("80"). **Строка в кавычках не должна содержать такие же кавычки внутри себя.** Так, если дата заключена в двойные кавычки, используйте одинарные кавычки для последующего заключения в кавычки, и наоборот. Вы можете также опустить кавычки для значений атрибутов, которые состоят только из следующих символов:

- символов английского алфавита (A - Z, a - z)
- цифр (0 - 9)
- промежутков времени
- дефисов (-), например HREF="<http://infofiz.ru/>". Существуют некоторые браузеры, которые допускают отсутствие кавычек или наличие элементов с открывающими кавычками без закрывающих. Однако, такую практику лучше не применять.

В пределах значения атрибута теги HTML не воспринимаются. А escape последовательности распознаются и интерпретируются, как символы.

Если значение атрибута такое же, как его имя, может быть использован **минимальный синтаксис** атрибута. То есть <UL COMPACT="COMPACT"> можно сократить до <UL COMPACT>. Некоторые браузеры даже требуют минимизации для некоторых атрибутов (COMPACT, ISMAP, CHECKED, NOWRAP, NOSHADE, NOHREF). Так что лучше использовать минимизированный синтаксис там, где это возможно.

Значения атрибутов должны быть отделены пробелами или незаполненными строками.

Что касается имен тегов, атрибутов и большинства значений атрибутов, то HTML нечувствителен к **регистру**. Вы можете, например, написать

TITLE, или Title, или title, или даже tLtLE, если Вам нравится. Но существуют и **чувствительные к регистру** конструкции языка, а именно:

- **escape** **последовательности** (более **официально** называемые *символьными объектами*), которые начинаются знаком & (например, <)

- **URL**, так как он может содержать наименования файлов, которые являются чувствительными к регистру во многих операционных системах (например, в Unix).

Когда документ выводится на экран, пробелы и пустые линии не сохраняются, за исключением текста, заключенного в теги PRE (предварительно отформатированный текст). То есть **любая последовательность пробелов, символов табуляции и пустых линий эквивалентна единственному пробелу в файле HTML**. С другой стороны, пробел в файле HTML может быть представлен с использованием любого количества пробелов или новыми (пустыми) строками.

Для того, чтобы текстовый файл превратился в HTML-файл, поменять его расширение с ".txt" на ".html" недостаточно. Надо соблюсти "**правило первой строки**":

Каждый HTML-документ, отвечающий спецификации HTML какой-либо версии, обязан начинаться **со строки декларации версии HTML <!DOCTYPE>**, которая обычно выглядит так:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//RU">

Эта строка поможет браузеру определить, как правильно интерпретировать данный документ. В данном случае мы говорим браузеру, что HTML соответствует международной спецификации версии 3.2, которая хоть и не отличается новизной, но, в отличие от более поздних версий, является полноценным, широко распространенным стандартом без каких-либо неопределенностей.

После объявления версии и типа документа необходимо обозначить его начало и конец. Это делается с помощью тэга-контейнера <HTML>.

Необходимо отметить, что любой HTML-документ открывается тэгом <HTML> и им же закрывается.

Затем, между тэгами <HTML> и </HTML> следует разместить заголовок документа (между тэгами <HEAD> и </HEAD>), и тело документа (между тэгами <BODY> и </BODY>).

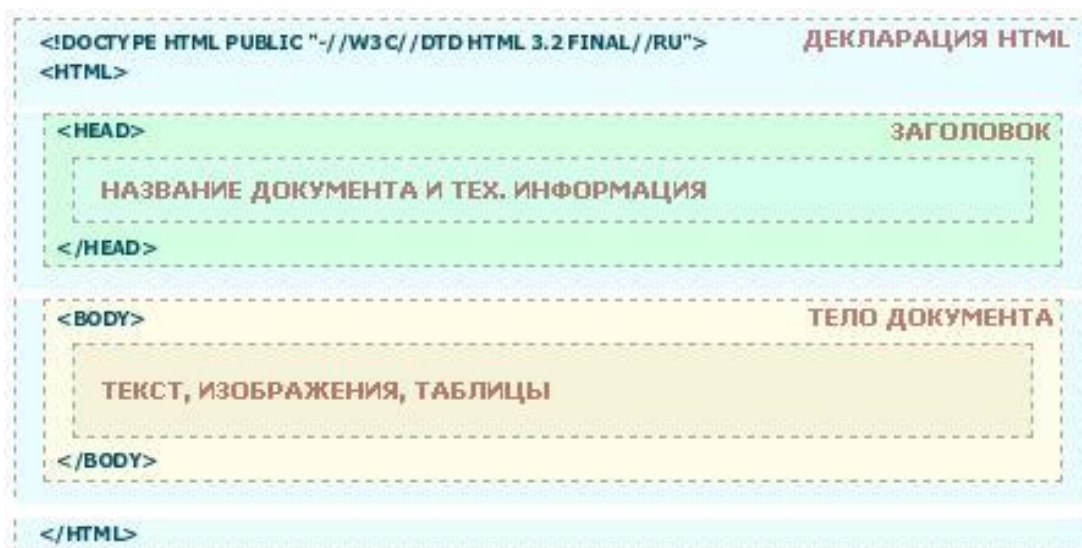
Между тэгами HEAD всегда должна находиться информация о кодировке страницы, например <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">, и теги определяющие название страницы <TITLE></TITLE>.

Только теперь между тэгами содержимого документа <BODY></BODY> можно располагать свою информацию, рисунки, видеофайлы, аудиофайлы...

Пример самого короткого HTML-документа:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//RU">
<HTML>
  <HEAD>
    <TITLE>Заголовок документа</TITLE>
  </HEAD>
  <BODY>
    Текст документа
  </BODY>
</HTML>
```

Если приведенный выше пример пояснить схематически, получится следующее:



Из схемы видно, что документ состоит из двух основных блоков - "заголовка" и "тела документа". Заголовок определяется с помощью элемента HEAD, а тело - элементом BODY.

Заголовок содержит "техническую" информацию о документе, хотя чаще всего используется только для обозначения его названия.

Тело документа - святая святых. Именно в нем находится все то, что отображается на странице: текст, картинки, таблицы. Соответственно, делаем вывод: большинство ваших HTML-экспериментов будет проводиться в пространстве между тэгами `<BODY>` и `</BODY>`