25.10.2021 МДК0301 Тема Введение в HTML

**Тема лекции Основные понятия и определения: теги парные и одиночные, ссылки, разметка HTML – документа.**

HTML (Hypertext Markup Language) - это код, который используется для структурирования и отображения веб-страницы и её контента. Например, контент может быть структурирован внутри множества параграфов, маркированных списков или с использованием изображений и таблиц данных. Как видно из названия, эта статья даст вам базовое понимание HTML и его функций.

**Что такое HTML на самом деле?**

HTML не является языком программирования; это *язык разметки*, и используется, чтобы сообщать вашему браузеру, как отображать веб-страницы, которые вы посещаете. Он может быть сложным или простым, в зависимости от того, как хочет веб-дизайнер. HTML состоит из ряда **элементов**, которые вы используете, чтобы вкладывать или оборачивать различные части контента, чтобы заставить контент отображаться или действовать определённым образом. Ограждающие теги могут сделать слово или изображение ссылкой на что-то ещё, могут сделать слова курсивом, сделать шрифт больше или меньше и так далее. Например, возьмём следующую строку контента:

Моя кошка очень раздражена

Если бы мы хотели, чтобы строка стояла сама по себе, мы могли бы указать, что это абзац, заключая его в теги абзаца:

<p>Моя кошка очень раздражена</p>

Анатомия HTML элемента

Давайте рассмотрим элемент абзаца более подробно.



Главными частями нашего элемента являются:

1. **Открывающий тег (Opening tag)**: Состоит из имени элемента (в данном случае, "p"), заключённого в открывающие и закрывающие **угловые скобки**.  Открывающий тег указывает, где элемент начинается или начинает действовать, в данном случае — где начинается абзац.
2. **Закрывающий тег (Closing tag):** Это то же самое, что и открывающий тег, за исключением того, что он включает в себя косую черту перед именем элемента. Закрывающий элемент указывает, где элемент заканчивается, в данном случае — где заканчивается абзац. Отсутствие закрывающего тега является одной из наиболее распространённых ошибок начинающих и может приводить к странным результатам.
3. **Контент (Content)**: Это контент элемента, который в данном случае является просто текстом.
4. **Элемент(Element)**: Открывающий тег, закрывающий тег и контент вместе составляют элемент.

Элементы также могут иметь атрибуты, которые выглядят так:



Атрибуты содержат дополнительную информацию об элементе, которую вы не хотите показывать в фактическом контенте. В данном случае, class это *имя* *атрибута,* а editor-note это *значение атрибута*. Класс позволяет дать элементу идентификационное имя, которое может позже использоваться, чтобы обращаться к элементу с информацией о стиле и прочих вещах.

Атрибут всегда должен иметь:

1. Пробел между ним и именем элемента (или предыдущим атрибутом, если элемент уже имеет один или несколько атрибутов).
2. Имя атрибута, за которым следует знак равенства.
3. Значение атрибута, заключённое с двух сторон в кавычки.

Вложенные элементы

Вы также можете располагать элементы внутри других элементов — это называется **вложением**. Если мы хотим заявить, что наша кошка **очень** раздражена, мы можем заключить слово "очень" в элемент <strong> , который указывает, что слово должно быть сильно акцентированно:

<p>Моя кошка <strong>очень</strong> раздражена.</p>

Вы, однако, должны убедиться, что ваши элементы правильно вложены: в примере выше мы открыли первым элемент <p>, затем элемент <strong>, потом мы должны закрыть сначала элемент <strong>, затем <p>. Приведённое ниже неверно:

<p>Моя кошка <strong>очень раздражена.</p></strong>

Элементы должны открываться и закрываться правильно, поэтому они явно располагаются внутри или снаружи друг друга. Если они перекрываются, как в примере выше, ваш веб-браузер будет пытаться сделать наилучшее предположение на основе того, что вы пытались сказать, что может привести к неожиданным результатам. Так что не стоит этого делать!

Пустые элементы

Некоторые элементы не имеют контента, и называются **пустыми элементами**. Возьмём  элемент <img>, который уже имеется в нашем HTML:

<img src="images/firefox-icon.png" alt="Моё тестовое изображение">

Он содержит два атрибута, но не имеет закрывающего тега </img>, и никакого внутреннего контента. Это потому, что элемент изображения не оборачивает контент для влияния на него. Его целью является вставка изображения в HTML страницу в нужном месте.

Анатомия HTML документа

Мы завершили изучение основ отдельных HTML элементов, но они не очень полезны сами по себе. Теперь мы посмотрим, как отдельные элементы объединяются в целую HTML страницу. Давайте вернёмся к коду, который мы запишем в наш html файл. Создадим изображение для браузера Мозилла.

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Моя тестовая страница</title>

 </head>

 <body>

 <img src="images/firefox-icon.png" alt="Моё тестовое изображение">

 </body>

</html>

Здесь мы имеем:

* <!DOCTYPE html> — доктайп. В прошлом, когда HTML был молод (около 1991/1992), доктайпы должны были выступать в качестве ссылки на набор правил, которым HTML страница должна была следовать, чтобы считаться хорошим HTML, что могло означать автоматическую проверку ошибок и другие полезные вещи. Однако в наши дни, никто не заботится об этом, и они на самом деле просто исторический артефакт, который должен быть включён для того, что бы все работало правильно. На данный момент это все, что вам нужно знать.
* <html></html> — элемент <html>. Этот элемент оборачивает весь контент на всей странице, и иногда известен как корневой элемент.
* <head></head> — элемент <head>. Этот элемент выступает в качестве контейнера для всего, что вы пожелаете включить на HTML страницу, но*не являющегося*контентом, который вы показываете пользователям вашей страницы. К ним относятся такие вещи, как ключевые слова и описание страницы, которые будут появляться в результатах поиска, CSS стили нашего контента, кодировка и многое другое.
* <body></body> — элемент <body>. В нем содержится *весь* контент, который вы хотите показывать пользователям, когда они посещают вашу страницу, будь то текст, изображения, видео, игры, проигрываемые аудиодорожки или что-то ещё.
* <meta charset="utf-8"> — этот элемент устанавливает UTF-8 кодировку вашего документа, которая включает в себя большинство символов из всех известных человечеству языков. По сути, теперь документ может обрабатывать любой текстовый контент, который вы в него вложите. Нет причин не устанавливать её, так как это может помочь избежать некоторых проблем в дальнейшем.
* <title></title> — элемент <title>. Этот элемент устанавливает заголовок для вашей страницы, который является названием, появляющимся на вкладке браузера загружаемой страницы, и используется для описания страницы, когда вы добавляете её в закладки/избранное.

**Изображения**

Давайте снова обратим наше внимание на элемент изображения:

<img src="images/firefox-icon.png" alt="Mоё тестовое изображение">

Как было сказано раньше, код встраивает изображение на нашу страницу в нужном месте. Это делается с помощью атрибута src (source, источник), который содержит путь к нашему файлу изображения.

Мы также включили атрибут alt (alternative, альтернатива). В этом атрибуте, вы указываете поясняющий текст для пользователей, которые не могут увидеть изображение, возможно, по следующим причинам:

1. У них присутствуют нарушения зрения. Пользователи со значительным нарушением зрения часто используют инструменты, называемые Screen Readers (экранные дикторы), которые читают для них альтернативный текст.
2. Что-то пошло не так, в результате чего изображение не отобразилось. Например, попробуйте намеренно изменить путь в вашем атрибуте src, сделав его неверным. Если вы сохраните и перезагрузите страницу, то вы должны увидеть что-то подобное вместо изображения:



Альтернативный текст - это "пояснительный текст". Он должен предоставить читателю достаточно информации, чтобы иметь представление о том, что передаёт изображение. В этом примере наш текст "My test image" ("Моё тестовое изображение") не годится. Намного лучшей альтернативой для нашего логотипа Firefox будет "The Firefox logo: a flaming fox surrounding the Earth" ("Логотип Firefox: огненный Лис вокруг Земли").

Сейчас попробуйте придумать более подходящий альтернативный текст для вашего изображения.

**Разметка текста**

В этом разделе рассмотрим некоторые из основных HTML элементов, которые вы будете использовать для разметки текста.

Заголовки

Элементы заголовка позволяют вам указывать определённые части вашего контента в качестве заголовков или подзаголовков. Точно так же, как книга имеет название, названия глав и подзаголовков, HTML документ может содержать то же самое. HTML включает шесть уровней заголовков <h1> (en-US)–<h6> (en-US), хотя обычно вы будете использовать не более 3-4 :

<h1>Мой главный заголовок</h1>

<h2>Мой заголовок верхнего уровня</h2>

<h3>Мой подзаголовок</h3>

<h4>Мой под-подзаголовок</h4>

Copy to Clipboard

Теперь попробуйте добавить подходящее название для вашей HTML страницы, чуть выше  элемента <img>.

Абзацы

Как было сказано раньше, элемент <p> предназначен для абзацев текста; вы будете использовать их регулярно при разметке текстового контента:

<p>Это одиночный абзац</p>

Добавьте свой образец текста в один или несколько абзацев, расположенных прямо под элементом <img>.

Списки

Большая часть веб-контента является списками и HTML имеет специальные элементы для них. Разметка списка всегда состоит по меньшей мере из двух элементов. Наиболее распространёнными типами списков являются нумерованные и ненумерованные списки:

1. **Ненумерованные списки** - это списки, где порядок пунктов не имеет значения, как в списке покупок. Они оборачиваются в элемент <ul>.
2. **Нумерованные списки -** это списки, где порядок пунктов имеет значение, как в рецепте. Они оборачиваются в элемент <ol>.

Каждый пункт внутри списков располагается внутри элемента <li> (list item, элемент списка).

Например, если мы хотим включить часть следующего фрагмента абзаца в список:

<p>Mozilla, мы являемся мировым сообществом технологов, мыслителей и строителей, работающих вместе ... </p>

Мы могли бы изменить разметку на эту:

<p>Mozilla, мы являемся мировым сообществом</p>

<ul>

 <li>технологов</li>

 <li>мыслителей</li>

 <li>строителей</li>

</ul>

<p>работающих вместе ... </p>

Copy to Clipboard

Попробуйте добавить упорядоченный или неупорядоченный список на свою страницу.

**Ссылки**

Ссылки очень важны — это то, что делает Интернет Интернетом. Чтобы добавить ссылку, нам нужно использовать простой элемент — <a> — *a* это сокращение от "anchor" ("якорь"). Чтобы текст в вашем абзаце стал ссылкой, выполните следующие действия:

1. Выберите некоторый текст. Мы выбрали текст "Манифест Mozilla".
2. Оберните текст в элемент <a>, например так:

<a>Манифест Mozilla</a>

1. Задайте элементу <a> атрибутhref, например так:

<a href="">Манифест Mozilla</a>

1. Заполните значение этого атрибута веб-адресом, на который вы хотите указать ссылку:

<a href="https://www.mozilla.org/ru/about/manifesto/details/">Манифест Mozilla</a>

Вы можете получить неожиданные результаты, если в самом начале веб-адреса вы опустите https:// или http:// часть, называемую *протоколом*. После создания ссылки, кликните по ней, чтобы убедиться, что она направляет вас туда, куда вы хотели.

href сначала может выглядеть довольно непонятым выбором для имени атрибута. Если у вас возникли проблемы с тем, чтобы запомнить его, можете запомнить, что атрибут href образуется как ***h****ypertext****ref****erence*("гипертекстовая ссылка").

Теперь добавьте ссылку на вашу страницу.

**Вопросы к домашнему заданию:**

1. **Что такое HTML?**
2. **Для чего служит HTML?**
3. **Где используеся HTML?**
4. **Что такое «теги»?**
5. **Для чего служат «теги»?**
6. **Где используются «теги»?**
7. **Что такое «контент»?**
8. **Что такое «элемент»?**

Присылать **kuzn117@yandex.ru Срок 26.10.2021**