**Конспект по html**

html - стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере.

**1. Структура html-документа:**

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="UTF-8">

<title>Моя первая страница</title>

</head>

<body></body>

</html>

Первая конструкция в любом HTML-документе — элемент <!DOCTYPE>. Он не относится к тегам и никаким образом не может отображаться на странице. Его задача — указать браузеру, какой стандарт HTML используется в этом документе.

Тег <html></html> является основой основ. Именно внутри него располагается вся информация. Благодаря этому тегу браузер понимает, где начинается контент, который необходимо обработать как HTML.

Тег <head></head> служит для хранения служебной информации. Здесь возможны самые разные сочетания тегов, которые подсказывают браузеру название страницы, описание, ключевые слова и так далее.

Метатег <meta> с атрибутом charset, он позволяет установить кодировку документа.

Для указания заголовка страницы используется специальный парный тег <title></title>, внутри которого указывается нужная информация.

Парный тег <body></body>, который является «телом» всей страницы. Именно здесь размещается вся информация, которая будет выведена на странице.

**2. Мета-теги**

Мета-теги используются для хранения информации предназначенной для браузеров и поисковых систем. Например, механизмы поисковых систем обращаются к мета-тегам для получения описания сайта, ключевых слов и других данных.

description - большинство поисковых серверов отображают содержимое поля description при выводе результатов поиска. Если этого тега нет на странице, то поисковый движок просто перечислит первые встречающиеся слова на странице, которые, как правило, оказываются не очень-то и в тему.

Автозагрузка - Чтобы автоматически загружать новый документ через определенный промежуток времени используется инструкция http-equiv="refresh".

Кодировка - Чтобы сообщить браузеру, в какой кодировке находятся символы веб-страницы, необходимо установить параметр charset.

**Пример:**

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

**3. Оформление текста**

Форматирование текста — средства его изменения, такие как выбор начертания шрифта и использование эффектов, позволяющих менять вид текста.

<b>Текст</b> - Жирное начертание текста

<i>Текст</i> - Курсивное начертание текста

<sup>Текст</sup> - Верхний индекс

<sub>Текст</sub> - Нижний индекс

<pre>Текст</pre> - Текст пишется как есть, включая все пробелы

**4. Списки**

Списки - используются для группировки связанных между собой фрагментов информации.

Существует три вида списков:

* маркированный список — <ul> — каждый элемент списка <li> отмечается маркером,
* нумерованный список — <ol> — каждый элемент списка <li> отмечается цифрой,
* список определений — <dl> — состоит из пар термин <dt> — <dd> определение.

**Пример:**

<ul>

<li><span>Спорт</span></li>

<li><span>Аниме</span></li>

<li><span>Видеоигры</span></li>

<li><span>Программирование</span></li>

</ul>

**5. Изображения**

HTML-изображения добавляются на веб-страницы с помощью элемента <img>. Использование графики делает веб-страницы визуально привлекательнее. Изображения помогают лучше передать суть и содержание веб-документа.

Элементы <map> и <area> позволяют создавать карты-изображения с активными областями.

Элемент <img> представляет изображение и его резервный контент, который добавляется с помощью атрибута alt.

Элемент <img> имеет обязательный атрибут src, значением которого является абсолютный или относительный путь к изображению.

**Пример:**

<img src="../jpg/IMG\_20190907\_101809.jpg" height="160" width="130" alt="">

**6. Гиперссылки**

Гиперссылки — это часть текста, которая переадресует на другой объект. Этот объект может быть текстом или изображением и может располагаться на этой же веб-странице. Также объектом может быть другой файл или страница, которые могут быть расположены как на том же компьютере, так и в общей сети.

Тег <a> является одним из важных элементов HTML и предназначен для создания ссылок. В зависимости от присутствия атрибутов name или href тег <a> устанавливает ссылку или якорь. Якорем называется закладка внутри страницы, которую можно указать в качестве цели ссылки. При использовании ссылки, которая указывает на якорь, происходит переход к закладке внутри веб-страницы.

**Пример:**

<a href="https://vk.com/id430508637">Мой ВК</a>

**7. Аудио-контент**

HTML-элемент <audio> используется для встраивания звукового контента в документ. Он может содержать один или более источников аудио, представленных с помощью атрибута src или элемента <source> – браузер выберет один наиболее подходящий. Он так же может предназначаться для потокового мультимедиа, используя интерфейс MediaStream.

**Атрибуты**

Autoplay Звук начинает играть сразу после загрузки страницы.

Controls - Добавляет панель управления к аудиофайлу.

Loop - Повторяет воспроизведение звука с начала после его завершения.

Preload - Используется для загрузки файла вместе с загрузкой веб-страницы.

Src - Указывает путь к воспроизводимому файлу.

**Пример:**

<audio controls src="../mp3/Killers\_with\_the\_cross.mp3"><a href="../mp3/Killers\_with\_the\_cross.mp3">Download audio</a></audio>

**8. Видео-контент**

Для встраивания видео контента в документ используйте элемент HTML <video>. Видео элемент может содержать один или несколько источников видео. Чтобы указать источник видео, необходимо использовать атрибут src или элемент <source>; браузер сам определит наиболее подходящий источник.

**Атрибуты**

Autoplay - Видео начинает воспроизводиться автоматически после загрузки страницы.

Controls - Добавляет панель управления к видеоролику.

Height - Задает высоту области для воспроизведения видеоролика.

Loop - Повторяет воспроизведение видео с начала после его завершения.

Poster - Указывает адрес картинки, которая будет отображаться, пока видео не доступно или не воспроизводится.

Preload - Используется для загрузки видео вместе с загрузкой веб-страницы.

Src - Указывает путь к воспроизводимому видеоролику.

Width - Задает ширину области для воспроизведения видеоролика.

**Пример:**

<video src="../mp4/Akame\_ga\_Kill!.mp4" width="480" height="270" poster="../jpg/91QjUH1o9jL.\_AC\_UF894,1000\_QL80\_.jpg" controls></video>

**9. Таблицы**

Элемент <table> служит контейнером для элементов, определяющих содержимое таблицы. Любая таблица состоит из строк и ячеек, которые задаются с помощью тегов <tr> и <td>. Внутри <table> допустимо использовать следующие элементы: <caption>, <col>, <colgroup>, <tbody>, <td>, <tfoot>, <th>, <thead> и <tr>.

Таблицы с невидимой границей долгое время использовались для верстки веб-страниц, позволяя разделять документ на модульные блоки. Подобный способ применения таблиц нашел воплощение на многих сайтах, пока ему на смену не пришел более современный способ верстки с помощью слоев.

**Атрибуты**

Align - Определяет выравнивание таблицы.

Background - Задает фоновый рисунок в таблице.

Bgcolor - Цвет фона таблицы.

Border - Толщина рамки в пикселах.

Bordercolor - Цвет рамки.

Cellpadding - Отступ от рамки до содержимого ячейки.

Cellspacing - Расстояние между ячейками.

Cols - Число колонок в таблице.

Frame - Сообщает браузеру, как отображать границы вокруг таблицы.

Height - Высота таблицы.

Rules - Сообщает браузеру, где отображать границы между ячейками.

Summary - Краткое описание таблицы.

Width - Ширина таблицы.

**Пример:**

<table border="1" cellpadding="4" cellspacing="0">

<tr>

<td colspan="2">Сёнэн</td>

<td>Исекай</td>

<td>Экшен</td>

<td>Психология</td>

</tr>

<tr>

<td>10</td>

<td>Bleach</td>

<td>Восхождение героя щита</td>

<td>КРД</td>

<td>Обещанный неверленд</td>

</tr>

<tr>

<td>9</td>

<td>Akame ga kill</td>

<td>Konosuba</td>

<td>Атака титанов</td>

<td>ERASED</td>

</tr>

<tr>

<td>8</td>

<td>Seven Deadly Sins</td>

<td>Overlord</td>

<td>Vinland Saga</td>

<td>Classroom of Elite</td>

</tr>

</table>

**10. Пользовательские формы**

Тег <form> устанавливает форму на веб-странице. Форма предназначена для обмена данными между пользователем и сервером. Область применения форм не ограничена отправкой данных на сервер, с помощью клиентских скриптов можно получить доступ к любому элементу формы, изменять его и применять по своему усмотрению.

Документ может содержать любое количество форм, но одновременно на сервер может быть отправлена только одна форма. По этой причине данные форм должны быть независимы друг от друга.

**Атрибуты**

accept-charset - Устанавливает кодировку, в которой сервер может принимать и обрабатывать данные.

Action - Адрес программы или документа, который обрабатывает данные формы.

Autocomplete - Включает автозаполнение полей формы.

Enctype - Способ кодирования данных формы.

Method - Метод протокола HTTP.

Name - Имя формы.

Novalidate - Отменяет встроенную проверку данных формы на корректность ввода.

Target - Имя окна или фрейма, куда обработчик будет загружать возвращаемый результат.

**Пример:**

<form action="#">

<p><i>Пожалуйста, заполните форму. Обязательные поля помечены </i><em>\*</em></p>

<fieldset>

<legend>Контактная информация</legend>

<label for="name">Имя <em>\*</em></label>

<input id="name"><br>

<label for="telephone">Телефон</label>

<input id="telephone"><br>

<label for="email">Email <em>\*</em></label>

<input id="email"><br>

</fieldset>

<fieldset>

<legend>Персональная информация</legend>

<label for="age">Возраст<em>\*</em></label>

<input id="age"><br>

<label for="gender">Пол</label>

<select id="gender">

<option value="female">Женщина</option>

<option value="male">Мужчина</option>

</select><br>

<label for="comments">Перечислите личные качества</label>

<textarea id="comments"></textarea>

</fieldset>

<fieldset>

<legend>Выберите ваших любимых животных</legend>

<label for="zebra"><input id="zebra" type="checkbox"> Зебра</label>

<label for="cat"><input id="cat" type="checkbox"> Кошак</label>

<label for="anaconda"><input id="anaconda" type="checkbox"> Анаконда</label>

<label for="human"><input id="human" type="checkbox"> Человек</label>

<label for="elephant"><input id="elephant" type="checkbox"> Слон</label>

<label for="wildebeest"><input id="wildebeest" type="checkbox"> Антилопа</label>

<label for="pigeon"><input id="pigeon" type="checkbox"> Голубь</label>

<label for="crab"><input id="crab" type="checkbox"> Краб</label>

</fieldset>

<p><input type="submit" value="Отправить информацию"></p>

</form>