HTML обозначает H Yper Т внутр M arkup L anguage и используется для создания структуры и содержания веб - страницы.

Большинство элементов HTML содержат открывающие и закрывающие теги с необработанным текстом или другими тегами HTML между ними.

Элементы HTML могут быть вложены в другие элементы. Вложенный элемент является дочерним по отношению к родительскому элементу.

Любой видимый контент должен быть помещен в открывающий и закрывающий <body>теги.

Заголовки и подзаголовки <h1>к <h6>тегам используются для увеличения текста.

<p>, <span>И <div>теги указать текст или блоки.

<em>И <strong>теги используются для выделения текста.

Разрывы строк создаются с помощью <br>тега.

Упорядоченные списки ( <ol>) нумеруются, а неупорядоченные списки ( <ul>) маркируются.

Изображения ( <img>) и видео ( <video>) могут быть добавлены путем ссылки на существующий источник.

<!DOCTYPE html>Декларация всегда должна быть первой строкой кода в HTML - файлах. Это позволяет браузеру знать, какую версию HTML ожидать.

<html>Элемент будет содержать все ваши HTML - кода.

Информация о веб-странице, как и заголовок, относится к <head>странице.

Вы можете добавить заголовок к своей веб-странице, используя <title>элемент внутри головы.

Заголовок веб-страницы отображается на вкладке браузера.

Якорные теги ( <a>) используются для ссылки на внутренние страницы, внешние страницы или контент на той же странице.

Вы можете создавать разделы на веб-странице и переходить к ним, используя <a>теги и добавляя ids к элементам, к которым вы хотите перейти.

Пробелы между элементами HTML помогают легче читать код, не изменяя внешний вид элементов в браузере.

Отступы также помогают облегчить чтение кода. Это делает отношения родитель-ребенок видимыми.

Комментарии написаны в HTML , используя следующий синтаксис: <!-- comment -->.

<table>Элемент создает таблицу.

<tr>Элемент добавляет строки в таблицу.

Чтобы добавить данные в строку, вы можете использовать <td>элемент.

Заголовки таблицы уточняют смысл данных. Заголовки добавляются с <th>элементом.

Данные таблицы могут охватывать столбцы с помощью colspanатрибута.

Данные таблицы могут охватывать строки с использованием rowspanатрибута.

Таблицы можно разделить на три основных раздела: заголовок, тело и нижний колонтитул.

Заголовок таблицы создается с помощью <thead>элемента.

Тело таблицы создается с помощью <tbody>элемента.

Нижний колонтитул таблицы создается с помощью <tfoot>элемента.

Все свойства CSS, которые вы узнали в этом курсе, могут быть применены к таблицам и их данным.

* Цель <form>состоит в том, чтобы позволить пользователям вводить информацию и отправлять ее.
* В <form>«S actionатрибут определяет , где информация формы идет.
* В <form>«S methodатрибут определяет , каким образом передается и обрабатывается информация.
* Чтобы добавить поля для пользователей для ввода информации, мы используем <input>элемент и устанавливаем typeатрибут в поле по нашему выбору:
* Установка typeдля "text"создает одно поле строки для ввода текста.
* Установка typeв "password"создает единственное поле строки, которое подвергает цензуре ввод текста.
* Установка typeдля "number"создает одно поле строки для ввода чисел.
* Установка typeдля "range"создает ползунок , чтобы выбрать из диапазона чисел.
* Установка typeдля "checkbox"создает один флажок , который может работать в паре с другими флажками.
* Установка typeна "radio"создает радиокнопку, которая может быть сопряжена с другими радиокнопками.
* Установка typeдля "list"будет на пару <input>с <datalist>элементом.
* Установка typeдля "submit"создает кнопку отправки.
* <select>Элемент заполняется <option>элементами и оказывает выпадающий список выбора.
* <datalist>Элемент заполняется <option>элементами и работает с <input>поиска с помощью выбора.
* <textarea>Элемент представляет собой поле ввода текста , который имеет настраиваемую область.
* Когда <form>отправляется, nameполя, принимающие ввод, и valueполя из них отправляются в виде name=valueпар.
* Использование этого <form>элемента в сочетании с другими элементами, перечисленными выше, позволяет нам создавать сайты, которые учитывают пожелания и потребности наших пользователей. Воспользуйтесь возможностью, чтобы взять то, что вы узнали, и применить это!
* Загрузите Forms: Cheat Sheet, чтобы помочь вам вспомнить содержание этого урока.
* Семантический HTML вводит значение для страницы через определенные элементы, которые обеспечивают контекст относительно того, что находится между тегами.
* Семантический HTML является современным стандартом и делает веб-сайт доступным для людей, которые используют программы чтения с экрана для перевода веб-страницы, и улучшает SEO вашего сайта.
* <header>, <nav>, <main>И <footer>создать базовую структуру веб - страницы.
* <section> определяет элементы в документе, такие как главы, заголовки или любые другие области документа с той же темой.
* <article> содержит контент, который имеет смысл сам по себе, такой как статьи, блоги, комментарии и т. д.
* <aside> содержит информацию, которая относится к основному контенту, но не требуется для понимания доминирующей информации.
* <figure> инкапсулирует все типы носителей.
* <figcaption>используется для описания СМИ в <figure>.
* <video>, <embed>и <audio>элементы используются для медиа-файлов.