**Раздел 1. Что такое HTML и как с ним работать**

* 1. **Структура HTML-документа**

HTML язык по своей сути не является языком программирования. Он является языком разметки гипертекстовых документов. Иными словами, он отвечает за расположение в документе Ваших текстов, рисунков, таблиц, предназначенных для жизни в сети Интернет. Заставить его посчитать, сколько будет дважды два невозможно, в нем нет логических функций, зато красиво и главное легко выложить информацию о том, что дважды два будет четыре — это запросто. Читается этот язык при помощи знакомых Вам программ, именуемых браузерами (обозревателями), которые «знают» стандартные команды html языка, и "пережевывая" их выводят на монитор компьютера документы в том виде, в котором хочет представить их веб мастер - составитель документа.

Рисунок 1 – Стандартная разметка HTML-документа

<!DOCTYPE html> <!-- Объявление формата документа -->

<html>

<head> <!-- Техническая информация о документе -->

<meta charset="UTF-8"> <!-- Определяем кодировку символов документа -->

<title>...</title> <!-- Задаем заголовок документа -->

<link rel="stylesheet" type="text/css" href="style.css"> <!-- Подключаем внешнюю таблицу стилей -->

<script src="script.js"></script> <!-- Подключаем сценарии -->

</head>

<body> <!-- Основная часть документа -->

</body>

</html>

Необходимо раз и навсегда запомнить, что если есть открывающий тег **<…>**, то обязательно должен быть и закрывающий **</…>** Хотя есть и исключения, как например тег **<br>** - он закрытия не требует, потому что говорит лишь о том, что следует писать с новой строки.

Чтобы разобраться во взаимодействии элементов веб-страницы, необходимо рассмотреть так называемые **«родственные отношения»** между элементами. Отношения между множественными вложенными элементами подразделяются на родительские и дочерние.

**Родительский элемент** — элемент, связанный с другими элементами более низкого уровня, и находящийся на дереве выше их.

**Дочерний элемент** — элемент, непосредственно подчиненный другому элементу более высокого уровня.

* + 1. Элемент <html>

Является корневым элементом документа. Все остальные элементы содержатся внутри <html>...</html>. Все, что находится за пределами элемента, не воспринимается браузером как HTML-код и никак им не обрабатывается.

* + 1. Элемент <head>

Раздел <head>...</head> содержит техническую информацию о странице: заголовок, описание, ключевые слова для поисковых машин, кодировку и т.д. Введенная в нем информация не отображается в окне браузера, однако содержит данные, которые указывают браузеру, как следует обрабатывать страницу.

* + 1. Элемент <title>

Обязательным элементом раздела <head> является <title>. Текст, размещенный внутри элемента <title>, отображается в строке заголовка веб-браузера.

Длина заголовка должна быть не более 60 символов, чтобы полностью поместиться в заголовке. Текст заголовка должен содержать максимально полное описание содержимого веб-страницы.

* + 1. Элемент <meta>

Необязательным элементом раздела <head> является элемент <meta>. С его помощью можно задать описание содержимого страницы и ключевые слова для поисковых машин, автора HTML-документа и прочие свойства метаданных.

Элемент <head> может содержать несколько элементов <meta>, потому что в зависимости от используемых атрибутов они несут различную информацию.

Описание содержимого страницы и ключевые слова одновременно можно указывать на нескольких языках, например, на русском и английском (рисунок 2):

Рисунок 2 – Применение элемента <meta>

<meta name="description" lang="ru" content="Описание содержимого страницы">

<meta name="description" lang="en" content="Description">

<meta name="keywords" lang="ru" content="Ключевые слова через запятую">

<meta name="keywords" lang="en" content="Keywords">

Для элемента <meta> доступны атрибуты, описанные в таблице 1:

Таблица 1 – Атрибуты элемента <meta>

|  |  |
| --- | --- |
| charset | Указывает кодировку символов для текущего HTML-документа: <meta charset="UTF-8"> |
| content | Содержит произвольный текст, который определяет значение, ассоциируемое с атрибутом http-equiv или name, в зависимости от их значения. |
| http-equiv | Контролирует действия браузера на данной веб-странице (эквивалент HTTP заголовков). При отображении страницы браузер будет следовать инструкциям, заданным в атрибуте: default-style указывает предпочтительный стиль для использования на странице. Атрибут content должен содержать идентификатор элемента <link>, который ссылается на таблицу стилей CSS, или идентификатор элемента <style>, содержащего таблицу стилей. refresh указывает время в секундах до перезагрузки страницы или время до перенаправления на другую страницу, если в атрибуте content после указания времени идет строка "url=адрес\_страницы". Автоматическая перезагрузка страницы через заданный промежуток времени, в данном примере, через 30 секунд: <meta http-equiv="refresh" content="30"> Если необходимо сразу перебросить посетителя на другую страницу, то можно указать URL-адрес в параметре url: <meta http-equiv="refresh" content="0; url=http://mail.ru/"> |
| name | Ассоциируется со значением, содержащемся в атрибуте content. Не должен использоваться в случае, если для элемента уже заданы атрибуты http-equiv, charset или itemprop. application-name указывает название веб-приложения, используемого на странице. author указывает имя автора документа в свободном формате. description определяет краткое описание к содержимому страницы, например: <meta name="description" content="Описание содержимого страницы"> generator указывает один из пакетов программного обеспечения, используемого для создания документа, например: <meta name="generator" content="WordPress 4.0"> . keywords содержит список ключевых слов, разделенных запятыми, соответствующих содержимому страницы, например: <meta name="keywords" content="Ключевые слова через запятую">. *Также атрибут name может принимать следующие значения из расширенной спецификации, такие как creator, googlebot, publisher, robots, slurp, viewport, хотя ни одно из них еще не было официально принято.* |

* + 1. Элемент <style>

Внутри этого элемента задаются стили, которые используются на странице. Для задания стилей в HTML-документе используется язык CSS. Таких элементов на странице может быть несколько.

* + 1. Элемент <link>

Задать стили для документа можно также при помощи другого способа — записать их в отдельный файл с расширением .css, например, style.css.

Подключить файл со стилями к веб-странице можно двумя способами:

- через директиву **@import url** (Рисунок 3)

- с использованием элемента **<link>**. Данный способ является предпочтительным (Рисунок 4).

Рисунок 3 – Подключение стилей через директиву @import url

<!DOCTYPE html>

<html>

<head>

<style>

@import url(style.css);

</style>

<meta>

<title> </title>

</head>

Рисунок 4 – Подключение стилей через <link>

<link rel="stylesheet" href="style.css" type="text/css">

* + 1. Элемент <script>

Элемент **<script>** позволяет присоединять к документу различные сценарии. Текст сценария может располагаться либо внутри этого элемента, либо во внешнем файле. Если текст сценария расположен во внешнем файле, то он подключается с помощью атрибутов элемента. Для элемента доступны атрибуты, приведенные в таблице 2.

Таблица 2 – Атрибуты элемента <script >

|  |  |
| --- | --- |
| async | Атрибут указывает на то, что сценарий будет загружаться асинхронно с остальной частью страницы (сценарий начнет выполняться после загрузки, если не указан defer). |
| charset | Определяет кодировку символов |
| crossorigin | Определяет, будет ли использоваться CORS при загрузке внешних скриптов (с использованием атрибута src). anonymous — перед загрузкой скрипта в кросс-доменный запрос браузер автоматически добавляет заголовок Origin, при этом не передаются параметры доступа (cookie, сертификат X.509, логин/пароль для базовой аутентификации по HTTP). Если в ответе сервера отсутствует заголовок Access-Control-Allow-Origin: имя домена, скрипт не будет загружен. use-credentials — перед загрузкой скрипта в кросс-доменный запрос браузер автоматически добавляет заголовок Origin с указанием параметров доступа (cookie, SSL-сертификат или пары логин/пароль). Если в ответе сервера отсутствует заголовок Access-Control-Allow-Credentials: true, скрипт не будет загружен. |
| defer | Интерпретация сценариев откладывается до окончания отображения документа на устройстве пользователя. |
| nonce | Обеспечивает безопасность, защищая от атак с внедрением межсайтового скриптинга (XSS, cross site scripting). Устанавливает правила использования встроенных скриптов с помощью nonce-значений и хэшей. Во время рендеринга страницы браузер для каждого инлайн-скрипта вычисляет хэши и сравнивает с перечисленными в CSP. Загрузка с ресурсов, не входящих в «белый список», блокируется. |
| src | Указывает на месторасположение файла со сценарием, значение атрибута — это url файла, содержащего JavaScript-программу. |
| type | Используются для объявления языка сценария, использованного при составлении содержимого элемента. |

* + 1. Элемент <body>

В разделе **<body>** располагается все содержимое документа.

* 1. **Краткий ликбез по основным HTML-тегам**

HTML-элементы — основа языка HTML. Каждый HTML-документ состоит из дерева HTML-элементов и текста. Каждый HTML-элемент обозначается начальным (открывающим) и конечным (закрывающим) тегом. Открывающий и закрывающий теги содержат имя элемента. Полное описание основных тегов представлено в Таблице 3.

Таблица 3 – Список HTML-элементов

|  |  |
| --- | --- |
| **Элемент** | **Описание** |
| <!DOCTYPE> | Объявляет тип документа и предоставляет основную информацию для браузера — его язык и версия. |
| <a> | Создаёт гипертекстовые ссылки. |
| <abbr> | Определяет текст как аббревиатуру или акроним. Поясняющий текст задаётся с помощью атрибута title. |
| <address> | Задает контактные данные автора/владельца документа или статьи. Отображается в браузере курсивом. |
| <area> | Представляет собой гиперссылку с текстом, соответствующей определенной области на карте-изображении или активную область внутри карты-изображения. Всегда вложен внутрь элемента <map>. |
| <article> | Раздел контента, который образует независимую часть документа или сайта, например, статья в журнале, запись в блоге, комментарий. |
| <aside> | Представляет контент страницы, который имеет косвенное отношение к основному контенту страницы/сайта. |
| <b> | Задает полужирное начертание отрывка текста, не придавая акцент или важность выделенному. |
| <blockquote> | Выделяет текст как цитату, применяется для описания больших цитат. |
| <body> | Представляет тело документа (содержимое, не относящееся к метаданным документа). |
| <br> | Перенос текста на новую строку. |
| <button> | Создает интерактивную кнопку. Элемент может содержать текст или изображение. |
| <canvas> | Холст-контейнер для динамического отображения изображений, таких как простые изображения, диаграммы, графики и т.п. Для рисования используется скриптовый язык JavaScript. |
| <caption> | Добавляет подпись к таблице. Вставляется сразу после открывающего тега <table>. |
| <col> | Выбирает для форматирования один или несколько столбцов таблицы, не содержащих информацию одного типа. |
| <colgroup> | Создает структурную группу столбцов, выделяющую множество логически однородных ячеек. |
| <data> | Элемент используется для связывания значения атрибута value, которое представлено в машиночитаемом формате и может быть обработано компьютером, с содержимым элемента. |
| <datalist> | Элемент-контейнер для выпадающего списка элемента <input>. Варианты значений помещаются в элементы <option>. |
| <del> | Помечает текст как удаленный, перечёркивая его. |
| <details> | Создаёт интерактивный виджет, который пользователь может открыть или закрыть. Представляет собой контейнер для контента, видимый заголовок виджета помещается в элемент <summary>. |
| <dfn> | Определяет слово как термин, выделяя его курсивом. Текст, идущий следом, должен содержать расшифровку этого термина. |
| <div> | Элемент-контейнер для разделов HTML-документа. Используется для группировки блочных элементов с целью форматирования стилями. |
| <dl> | Элемент-контейнер, внутри которого находятся термин и его описание. |
| <em> | Выделяет важные фрагменты текста, отображая их курсивом. |
| <embed> | Элемент-контейнер для встраивания внешнего интерактивного контента или плагина. |
| <fieldset> | Группирует связанные элементы в форме, рисуя рамку вокруг них. |
| <figure> | Самодостаточный элемент-контейнер для такого контента как иллюстрации, диаграммы, фотографии, примеры кода, обычно с подписью. |
| <footer> | Определяет завершающую область (нижний колонтитул) документа или раздела. |
| <form> | Форма для сбора и отправки на сервер информации от пользователей. Не работает без атрибута action. |
| <h1-h6> | Создают заголовки шести уровней для связанных с ними разделов. |
| <head> | Элемент-контейнер для метаданных HTML-документа, таких как <title>, <meta>, <script>, <link>, <style>. |
| <header> | Секция для вводной информации сайта или группы навигационных ссылок. Может содержать один или несколько заголовков, логотип, информацию об авторе. |
| <hr> | Горизонтальная линия для тематического разделения параграфов. |
| <html> | Корневой элемент HTML-документа. Сообщает браузеру, что это HTML-документ. Является контейнером для всех остальных html-элементов. |
| <i> | Выделяет отрывок текста курсивом, не придавая ему дополнительный акцент. |
| <iframe> | Создает встроенный фрейм, загружая в текущий HTML-документ другой документ. |
| <img> | Встраивает изображения в HTML-документ с помощью атрибута src, значением которого является адрес встраиваемого изображения. |
| <input> | Создает многофункциональные поля формы, в которые пользователь может вводить данные. |
| <ins> | Выделяет текст подчеркиванием. Применяется для выделения изменений, вносимых в документ. |
| <kbd> | Выделяет текст, который должен быть введён пользователем с клавиатуры, шрифтом семейства monospace. |
| <label> | Добавляет текстовую метку для элемента <input>. |
| <legend> | Заголовок элементов формы, сгруппированных с помощью элемента <fieldset>. |
| <li> | Элемент маркированного или нумерованного списка. |
| <link> | Определяет отношения между документом и внешним ресурсом. Также используется для подключения внешних таблиц стилей. |
| <main> | Контейнер для основного уникального содержимого документа. На одной странице должно быть не более одного элемента <main>. |
| <map> | Создаёт активные области на карте-изображении. Является контейнером для элементов <area>. |
| <meta> | Используется для хранения дополнительной информации о странице. Эту информацию используют браузеры для обработки страницы, а поисковые системы — для ее индексации. В блоке <head> может быть несколько элементов <meta>, так как в зависимости от используемых атрибутов они несут разную информацию. |
| <nav> | Раздел документа, содержащий навигационные ссылки по сайту. |
| <noscript> | Определяет секцию, не поддерживающую сценарий (скрипт). |
| <object> | Контейнер для встраивания мультимедиа (например, аудио, видео, Java-апплеты, ActiveX, PDF и Flash). Также можно вставить другую веб-страницу в текущий HTML-документ. Для передачи параметров встраиваемого плагина используется элемент <param>. |
| <ol> | Упорядоченный нумерованный список. Нумерация может быть числовая или алфавитная. |
| <optgroup> | Контейнер с заголовком для группы элементов <option>. |
| <option> | Определяет вариант/опцию для выбора в раскрывающемся списке <select>, <optgroup> или <datalist>. |
| <output> | Поле для вывода результата вычисления, рассчитанного с помощью скрипта. |
| <p> | Параграфы в тексте. |
| <param> | Определяет параметры для плагинов, встраиваемых с помощью элемента <object>. |
| <picture> | Элемент-контейнер, содержащий один элемент <img> и ноль или несколько элементов <source>. Сам по себе ничего не отображает. Дает возможность браузеру выбирать наиболее подходящее изображение. |
| <pre> | Выводит текст без форматирования, с сохранением пробелов и переносов текста. Может быть использован для отображения компьютерного кода, сообщения электронной почты и т.д. |
| <progress> | Индикатор выполнения задачи любого рода. |
| <q> | Определяет краткую цитату. |
| <ruby> | Контейнер для Восточно-Азиатских символов и их расшифровки. |
| <rb> | Определяет вложенный в него текст как базовый компонент аннотации. |
| <rt> | Добавляет краткую характеристику сверху или снизу от символов, заключенных в элементе <ruby>, выводится уменьшенным шрифтом. |
| <rtc> | Отмечает вложенный в него текст как дополнительную аннотацию. |
| <s> | Отображает текст, не являющийся актуальным, перечеркнутым. |
| <script> | Используется для определения сценария на стороне клиента (обычно JavaScript). Содержит либо текст скрипта, либо указывает на внешний файл сценария с помощью атрибута src. |
| <section> | Определяет логическую область (раздел) страницы, обычно с заголовком. |
| <select> | Элемент управления, позволяющий выбирать значения из предложенного множества. Варианты значений помещаются в <option>. |
| <small> | Отображает текст шрифтом меньшего размера. |
| <source> | Указывает местоположение и тип альтернативных медиаресурсов для элементов <picture>, <video>, <audio>. |
| <span> | Контейнер для строчных элементов. Можно использовать для форматирования отрывков текста, например, выделения цветом отдельных слов. |
| <strong> | Расставляет акценты в тексте, выделяя полужирным. |
| <style> | Подключает встраиваемые таблицы стилей. |
| <sub> | Задает подстрочное написание символов, например, индекса элемента в химических формулах. |
| <summary> | Создаёт видимый заголовок для элемента <details>. Отображается с закрашенным треугольником, кликнув по которому можно просмотреть подробности заголовка. |
| <sup> | Задает надстрочное написание символов. |
| <table> | Элемент для создания таблицы. |
| <tbody> | Определяет тело таблицы. |
| <td> | Создает ячейку таблицы. |
| <template> | Используется для объявления фрагментов HTML-кода, которые могут быть клонированы и вставлены в документ скриптом. Содержимое элемента не является его дочерним элементом. |
| <textarea> | Создает большие поля для ввода текста. |
| <tfoot> | Определяет нижний колонтитул таблицы. |
| <th> | Создает заголовок ячейки таблицы. |
| <thead> | Определяет заголовок таблицы. |
| <time> | Определяет дату/время. |
| <title> | Заголовок HTML-документа, отображаемый в верхней части строки заголовка браузера. Также может отображаться в результатах поиска, поэтому это следует принимать во внимание предоставление названия. |
| <tr> | Создает строку таблицы. |
| <track> | Добавляет субтитры для элементов <audio> и <video>. |
| <u> | Выделяет отрывок текста подчёркиванием, без дополнительного акцента. |
| <ul> | Создает маркированный список. |