# МАТЕРИАЛ

## HTML

### Атрибуты

Атрибуты задаются в начальном тэге элемента и состоят из имени и значения, которые отделяются друг от друга знаком равно (=).

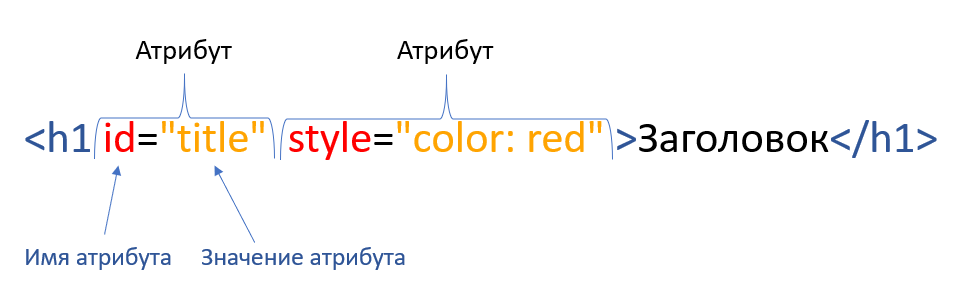
Например в атрибуте align="center" **align** является именем, а **center** значением.

Рассмотрим следующий пример:

| <p align="center">Текст выровненный по центру</p> |
| --- |

В этом примере используется тег <p>, который предназначен для обозначения абзаца. В начальном теге элемента дополнительно был добавлен атрибут. Этот атрибут содержит значение, указывающее на то, как нужно выровнять текст.

HTML-теги могут содержать один или несколько атрибутов. Атрибуты добавляются в тег для того, чтобы информировать браузер о том, как данный тег должен отображаться в html-документе.



### Перевод строки

Для принудительного разрыва строк используется одинарный тег <br>.

Если в HTML-документе набрать текст:

<p>

Строка1  
 Строка2  
 Строка3

</p>

то Web-браузер отобразит его в одну строку: Строка1 Строка2 Строка3. Для того чтобы строки располагались друг под другом, необходимо добавить тег <br> в местах, где нужно перенести строку:

| <p>  Строка1<br>  Строка2<br>  Строка3  </p> |
| --- |

### Авторское форматирование

Элемент HTML <pre> представляет собой предварительно отформатированный текст, который должен быть представлен точно так, как написано в HTML-файле. Текст обычно отображается с использованием непропорционального ("monospace") шрифта. Пробелы внутри этого элемента отображаются как записанные.

| <pre>  Время - начинаю про Ленина рассказ.  Но не потому,  что горя нету более,  время потому,  что резкая тоска  стала ясною  осознанною болью </pre> |
| --- |

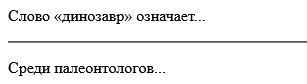
Данный элемент не имеет специальных атрибутов.

### Горизонтальная линия

Тег <hr> - та самая горизонтальная черта, которая разделяет смысловые блоки на странице. Он помогает отделить независимые друг от друга блоки — например, подразделы в статье. Это блочный элемент, поэтому он встаёт в отдельном ряду, визуально разделяя информацию на странице.

| <p>Слово «динозавр» означает...</p> <hr> <p>Среди палеонтологов...</p> |
| --- |

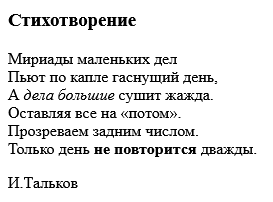
Вот, что мы видим в браузере:



### Практика

#### Задание 1

А теперь давайте немного попрактикуемся, создайте html-файл (кодировка utf-8) с заголовком "Задача 1", результат которого показан на рисунке. Используйте подходящий тег заголовков (<h1>-<h6>), теги <br>, <p>, <i>, <em>, <strong>, <b> (необязательно все перечисленные). Выясните различие тегов <b> и <strong>, тегов <i> и <em>. В html-код добавьте комментарии: дату решения данной задачи и ФИО.



***Вариант решения:***

| <!DOCTYPE html> <html>  <head>  <!-- 06.11.2022 -->  <title>Стихотворение</title>  </head>  <body>  <h3>Стихотворение</h3>  <p>  Мириады маленьких дел<br>  Пьют по капле гаснущий день,<br>  А <i>дела большие</i> сушит жажда.<br>  Оставляя все на «потом».<br>  Прозреваем задним числом.<br>  Только день <b>не повторится</b> дважды.  </p>  <!-- Степан Петров -->  <p>И.Тальков</p>  </body> </html> |
| --- |

***Примечание:***

*При решении этой задачи, можно заменить теги <i> на <em> и <b> на <strong>, результат будет тотже. Комментарии допустимо оставить в любом удобном месте, не нарушив при этом разметку.*

*Теги <b> и <strong> и теги <i> и <em> - внешне ничем не отличаются.*

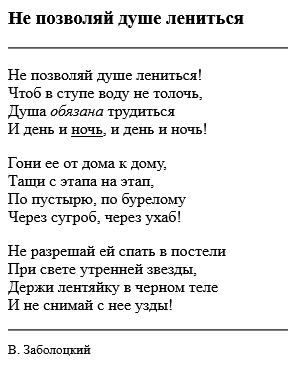
*Тег <em> определяет текст, на который сделан особый акцент, меняющий смысл предложения.*

*Тег <i> применяется для обозначения текста, который отличается от окружающего текста, но не является более важным.*

*С тегами <b> и <strong> ситуация та же.*

#### Задание 2

Создайте html-файл (кодировка utf-8) с заголовком "Задача 2", результат которого показан на рисунке. Используйте подходящий тег заголовков (h1-h6), теги <br>, <p>, <i>, <em>, <strong>, <b>, <hr>, <small> (необязательно все перечисленные). Обратите внимание, что шрифт фамилии автора меньшего размера. Заметим, что тег <hr> в разных браузерах может отображаться по-разному. В html-код добавьте условие данной задачи в виде комментариев.



***Вариант решения:***

| <!DOCTYPE html> <html>  <head>  <title>Не позволяй душе лениться</title>  </head>  <body>  <h3>Не позволяй душе лениться</h3>  <hr>  <p>  Не позволяй душе лениться!<br>  Чтоб в ступе воду не толочь,<br>  Душа <em>обязана</em> трудиться<br>  И день и <u>ночь</u>, и день и ночь!  </p>  <p>  Гони ее от дома к дому,<br>  Тащи с этапа на этап,<br>  По пустырю, по бурелому<br>  Через сугроб, через ухаб!  </p>  <p>  Не разрешай ей спать в постели<br>  При свете утренней звезды,<br>  Держи лентяйку в черном теле<br>  И не снимай с нее узды!  </p>  <hr>  <small>В. Заболоцкий</small>  <!--  создайте html-файл (кодировка utf-8) с заголовком "Задача 2", результат которого показан на рисунке. Используйте подходящий тег заголовков (h1-h6), теги <br>, <p>, <i>, <em>, <strong>, <b>, <hr>, <small> (необязательно все перечисленные). Обратите внимание, что шрифт фамилии автора меньшего размера. Заметим, что тег <hr> в разных браузерах может отображаться по-разному. В html-код добавьте условие данной задачи в виде комментариев.  -->  </body> </html> |
| --- |

### Изображения

Изображения вставляются в Web-страницы с помощью одинарного тега <img>

Элемент <img> представляет изображение и его резервный контент, который добавляется с помощью атрибута alt. Так как элемент <img> является строчным, то рекомендуется располагать его внутри блочного элемента, например, <p> или <div>.

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

| <img src="foto.jpg" alt="Текст подсказки"> |
| --- |

Атрибут alt добавляет альтернативный текст для изображения. Выводится на месте появления изображения до его загрузки или при отключенной графике.

Синтаксис: alt="описание изображения".

Использование графики делает веб-страницы визуально привлекательнее. Изображения помогают лучше передать суть и содержание веб-документа.

### Списки

Списки представляют собой упорядоченный набор значений, который помогает нам систематизировать важную информацию, делая ее более доступной для понимания. Язык гипертекстовой разметки HTML позволяет составлять списки по следующим правилам:

* Нумерованный (упорядоченный) список.
* Маркированный (неупорядоченный) список.
* Список описаний.

#### Нумерованный список

Давайте рассмотрим примеры использования:

| <!DOCTYPE html> <html> <head>  <title>Пример использования тега <ol></title> </head> <body>  <ol>  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl> </body> </html> |
| --- |

Выглядеть на странице это будет соответственно так:

1. Первый пункт.
2. Второй пункт.
3. Третий пункт.

Если вы хотите, чтобы список начинался с определённого номера (не с 1), то необходимо указать атрибут start для тега <ol>.

Например:

| <ol start="101"> <!-- список начнётся с номера 101 --> |
| --- |

Выглядеть на странице это будет соответственно так:

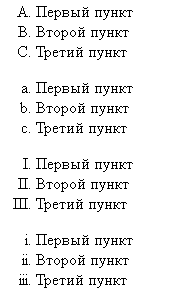
1. Первый пункт.
2. Второй пункт.
3. Третий пункт.

Ещё один интересный атрибут - type, который позволит Вам задать буквенную нумерацию ("A" – большие, "a" – строчные), либо нумерацию из римских цифр ("I" – в верхнем регистре, "i" – в нижнем регистре).

Рассмотрим пример в котором представлены все возможные значения атрибута type (отличные от значения по умолчанию):

| <!DOCTYPE html> <html>  <head>  <title>Пример использования атрибута type HTML тега <оl></title>  </head>  <body>  <!--указываем, что список формируется в алфавитном порядке заглавными буквами -->  <ol type = "A">  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  <!--указываем, что список формируется в алфавитном порядке строчными буквами -->  <ol type = "a">  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  <!--указываем, что список формируется римскими цифрами в верхнем регистре -->  <ol type = "I">  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  <!--указываем, что список формируется римскими цифрами в нижнем регистре -->  <ol type = "i">  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  </body> </html> |
| --- |
|  |

Результат нашего примера:



Еще один нюанс: при использовании атрибута start с буквами (type = "A" и type = "a"), число, указанное в значении атрибута является порядковым номером буквы в алфавите. Например, start = "4", будет соответствовать букве "D" и список начнётся именно с неё. При использовании значения start = "27" счетчик обнуляется, при этом список становится двухзначным ("27" = "AA", "28" = "AB", "29" = "AC"...). Значения должны быть целочисленными, допускается использование отрицательных значений.

Обращаю Ваше внимание, что допускается формировать нумерованные (упорядоченные) списки, вложенные в другие нумерованные списки (внутри элемента списка <li>):

| <!DOCTYPE html> <html>  <head>  <title>Пример нумерованного списка, вложенного в другой нумерованный список</title>  </head>  <body>  <ol>  <li>Первый пункт  <ol> <!--открываем новый нумерованный список внутри элемента списка-->  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  </li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl>  </body> </html> |
| --- |

Выглядеть на странице это будет соответственно так:

1. Первый пункт.
   1. Первый пункт.
   2. Второй пункт.
   3. Третий пункт.
2. Второй пункт.
3. Третий пункт.

Но и это еще не все, атрибут **reversed** элемента позволяет задать, что порядок в нумерованном (упорядоченном) списке должен идти **по убыванию**. Атрибут не поддерживается браузерами Internet Explorer и Edge.

| <ol reversed>  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </оl> |
| --- |

#### Маркированный список

Маркированный (неупорядоченный) список в HTML определяется тегом <ul> (Unordered List Element). Каждый элемент списка должен начинаться с тега <li> (сокращенное от английского list item - элемент списка).

По умолчанию, маркированные списки отображаются на веб-странице в виде списка, начинающегося с маленького чёрного круга.

Давайте рассмотрим примеры использования:

| <!DOCTYPE html> <html>  <head>  <title>Пример использования элемента <ul></title>  </head>  <body>  <ul>  <li>Светлое</li>  <li>Тёмное</li>  <li>Пятница</li>  </ul>  </body> </html> |
| --- |

Выглядеть на странице это будет соответственно так:

* Светлое
* Тёмное
* Пятница

*Забегая вперед, скажу, что каждый элемент HTML имеет стиль по умолчанию. Изменение стиля по умолчанию в HTML элементе может быть сделано с использованием глобального атрибута style. Более подробно этот атрибут будет рассмотрен в позднее в рамках изучения CSS.*

*Для изменения типа маркера (стиля) мы можете воспользоваться свойством CSS list-style-type, чтобы определить стиль маркера.*

Обращаю Ваше внимание, что допускается формировать маркированные (неупорядоченные) списки, вложенные в другие маркированные списки (внутри элемента списка <li>) при этом элементы вложенного списка по умолчанию будут с пустым кругом внутри, a последующие будут с квадратом.

#### Список описаний

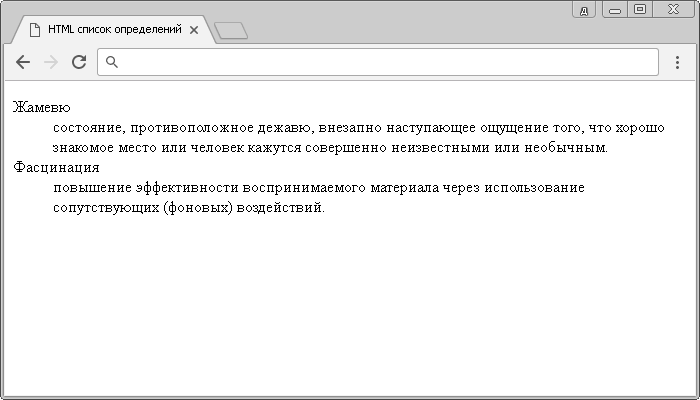
Списки описаний используются для формирования пар типа «имя/значение» / «вопрос/ответ» и т.п.

Тэг <dl> (HTML Description List Element) определяет список, тэг <dt> (HTML Definition Term Element) определяет его имя, а тег <dd> (HTML Description Element) описывает его значение.

Рассмотрим пример использования:

| <!DOCTYPE html> <html>  <head>  <title>Пример использования списка описаний</title>  </head>  <body>  <dl>  <dt>Жамевю</dt>  <dd>состояние, противоположное дежавю, внезапно наступающее ощущение того, что хорошо знакомое место  или человек кажутся совершенно неизвестными или необычным.</dd>  <dt>Фасцинация</dt>  <dd>повышение эффективности воспринимаемого материала через использование сопутствующих (фоновых) воздействий.</dd>  </dl>  </body> </html> |
| --- |

Как вы можете заметить значение списка описаний (элемент <dd>) имеет по умолчанию внешний отступ с левой стороны равный 40 пикселям:



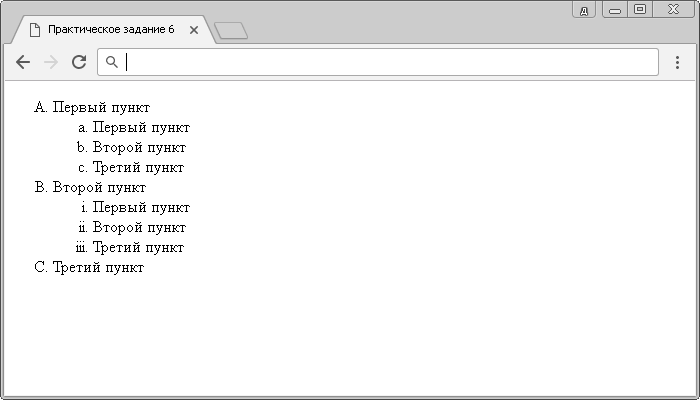
Обращаю Ваше внимание, что допускается формировать списки описаний, вложенные в другие списки описаний (внутри элемента значения списка <dd>).

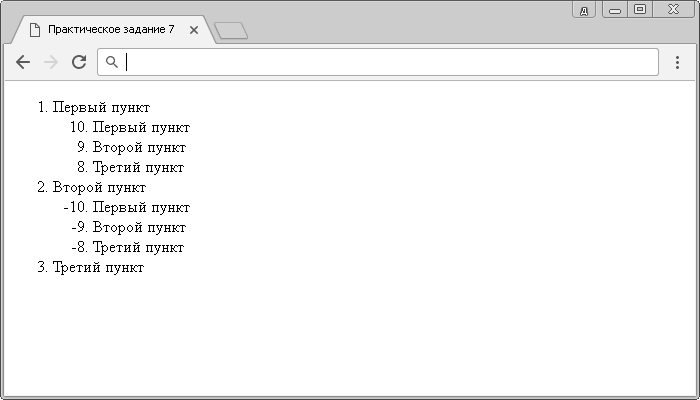
*Маркированные (неупорядоченные), нумерованные (упорядоченные) и списки описаний внутри себя могут содержать элементы нового списка (внутри элементов списка / значений списка), другие элементы HTML, такие как, изображения, ссылки и т.д. Списки можно размещать горизонтально, формировать из них меню навигации и так далее. Более подробно будем рассматривать позднее.*

### Практика

#### Задание 3

Используя полученные знания составьте следующие HTML списки:





**Решение:**

<ol type="A">  
 <li>Первый пункт  
 <ol type="a">  
 <li>Первый пункт</li>  
 <li>Второй пункт</li>  
 <li>Третий пункт</li>  
 </ol>  
 </li>  
 <li>Второй пункт  
 <ol type="i">  
 <li>Первый пункт</li>  
 <li>Второй пункт</li>  
 <li>Третий пункт</li>  
 </ol>  
 </li>  
 <li>Третий пункт</li>  
 </оl>

| <ol>  <li>Первый пункт  <ol start="10" reversed>  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </ol>  </li>  <li>Второй пункт  <ol start="-10">  <li>Первый пункт</li>  <li>Второй пункт</li>  <li>Третий пункт</li>  </ol>  </li>  <li>Третий пункт</li>  </оl> |
| --- |

### Специальные символы

В тексте HTML-документа разрешено применять не все символы, которые вы видите на клавиатуре. Впрочем, это логично. Если каждый тег начинается с символа (<) и заканчивается символом (>), то вполне логично, что эти символы нельзя указывать внутри обычного текста. Эти символы имеют особое значение в HTML-документах. Когда браузер их видит он думает, что это HTML-теги и пытается интерпретировать их как таковые.

Для решения проблемы с добавлением символов, запрещенных к вводу в тексте HTML-документа, был разработан набор спецсимволов – последовательностей, которые заменяют запрещенные для ввода с клавиатуры символы. Символы, которые не присутствуют на вашей клавиатуре, также могут быть заменены спецсимволами. Каждая такая последовательность (ссылка-мнемоника) обязательно начинается с символа амперсанда (&) и заканчиваются точкой с запятой (;). Между (&) и (;) можно ввести:

* имя спецсимвола (&имя;)
* числовой код, который вводится после знака (#), то есть при наборе спецсимвола путем указания его числового кода нужно использовать следующую запись: (&#код;)

Числовой код может быть представлен в десятичном или шестнадцатеричном виде.

Например, чтобы отобразить знак меньше (<) мы должны написать: &lt; (имя символа) или &#60; (числовой код в десятичной системе) или &#x0003C; (числовой код в шестнадцатеричной системе).

Перечень некоторых популярных спецсимволов HTML:

| **Символ** | **Описание** | **Мнемоническое имя** | **Числовой код** |
| --- | --- | --- | --- |
|  | неразрывный пробел | &nbsp; | &#160; |
| < | меньше чем | &lt; | &#60; |
| > | больше чем | &gt; | &#62; |
| & | амперсанд | &amp; | &#38; |
| " | прямая двойная кавычка | &quot; | &#34; |
| ' | апостроф | &apos; | &#39; |
| « | левая двойная угловая кавычка | &laquo; | &#171; |
| » | правая двойная угловая кавычка | &raquo; | &#187; |
| ‘ | левая одинарная кавычка | &lsquo; | &#8216; |
| ’ | правая одинарная кавычка | &rsquo; | &#8217; |
| ‚ | нижняя одинарная кавычка | &sbquo; | &#8218; |
| “ | левая двойная кавычка | &ldquo; | &#8220; |
| ” | правая двойная кавычка | &rdquo; | &#8221; |
| „ | нижняя двойная кавычка | &bdquo; | &#8222; |
| € | euro | &euro; | &#8364; |
| © | копирайт | &copy; | &#169; |
| ® | знак зареrистрированной торrовой марки | &reg; | &#174; |

### 

### 

### Самостоятельная работа

Создайте html-файл (кодировка utf-8) с заголовком "Задача 4", результат которого показан на рисунке. Используйте все полученные знания для достижения результата.



**Вариант решения:**

| <!DOCTYPE html> <html>  <head>  <title>Зачем нужны изображения на сайтах?</title>  </head>  <body>  <h1>Зачем использовать изображения на сайтах?</h1>  <img src="Lab\_03\_img.jpg" alt="Очень важная картинка">  <dl>  <dt><h4>Статистика</h4></dt>  <dd>  Согласно статистике, примерно 65% людей во всем мире<br>  являются «визуально воспринимаемыми» — они воспринимают,<br>  запоминают и обучаются на изображениях и графике,<br>  вы знаете, почему я начинаю с этого, верно?  </dd>  <dt><h4>Научный взгляд</h4></dt>  <dd>  Научно доказано, что человеческий мозг<br>  обрабатывает визуальную информацию <strong>в 60 000 раз быстрее</strong>, чем текстовое содержимое!  </dd>  </dl>  <hr>  <p>2022 &copy; Рога и Копыта</p>  </body> </html> |
| --- |

*Вполне допустимо решить задачу несколько иным способом, если визуально результат близок к тому, что на изображении.*

# 

# ШПАРГАЛКА

## HTML

| **Элемент** | **Описание** |
| --- | --- |
| <!--...--> | Используется для добавления комментариев. |
| <!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. |
| <audio> | Загружает звуковой контент на веб-страницу. |
| <b> | Задает полужирное начертание отрывка текста, не придавая акцент или важность выделенному. |
| <base> | Задает базовый адрес (URL), относительно которого вычисляются все относительные адреса. Это поможет избежать проблем при переносе страницы в другое место, так как все ссылки будут работать, как и прежде. |
| <bdi> | Изолирует отрывок текста, написанный на языке, в котором чтение текста происходит справа налево, от остального текста. |
| <bdo> | Отображает текст в направлении, указанном в атрибуте dir, переопределяя текущее направление написания текста. |
| <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>. |
| <cite> | Используется для указания источника цитирования. Отображается курсивом. |
| <code> | Представляет фрагмент программного кода, отображается шрифтом семейства monospace. |
| <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>. |
| <dd> | Используется для описания термина из элемента <dt>. |
| <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. |
| <dialog> | Интерактивный элемент, с которым взаимодействует пользователь для выполнения задачи, например, диалоговое окно, инспектор или окно. Без атрибута open не виден для пользователя. |
| <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. |
| <dt> | Используется для задания термина. |
| <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. |
| <figcaption> | Заголовок/подпись для элемента <figure>. |
| <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>. |
| <mark> | Выделяет фрагменты текста, помечая их желтым фоном. |
| <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. |
| <meter> | Индикатор измерения в заданном диапазоне. |
| <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. |
| <rp> | Выводит альтернативный текст в случае если браузер не поддерживает элемент <ruby>. |
| <s> | Отображает текст, не являющийся актуальным, перечеркнутым. |
| <samp> | Используется для вывода текста, представляющего результат выполнения программного кода или скрипта, а также системные сообщения. Отображается моноширинным шрифтом. |
| <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. |
| <var> | Выделяет переменные из программ, отображая их курсивом. |
| <video> | Добавляет на страницу видео-файлы. Поддерживает 3 видео формата: MP4, WebM, Ogg. |
| <wbr> | Указывает браузеру возможное место разрыва длинной строки. |

### 

### 

## CSS

### Текст

| text-align: center; | Выравнивание текста по центру; |
| --- | --- |
| text-align: left; | Выравнивание текста по левому краю; |
| text-align:right; | Выравнивание текста по правому краю; |
| text-decoration: none; | Текст без подчёркивания; |
| text-decoration: underline; | Подчёркнутый текст; |
| text-decoration: overline; | Черта над текстом; |
| text-decoration: line-through; | Зачёркнутый текст; |
| text-decoration: blink; | Мигающий текст; |
| text-transform: uppercase; | Делает все буквы заглавными; |
| text-transform: capitalize; | Делает все первые буквы заглавными; |
| text-indent: 30px; | Красная строка; |
| word-spacing: 30px; | Задаёт расстояние между словами; |
| letter-spacing:2px; | Задаёт расстояние между буквами; |
| line-height: 10px; | Задаёт фиксированную высоту строки; |
| direction: rtl; | Задаёт направление; текста справа налево. |
| color: #00ff00; | Цвет текста; |
| text-schadow: | Тень текста; |

### Шрифт

| font-style: italic; | Шрифт курсив; |
| --- | --- |
| font-weight: bold; | Жирный шрифт; |
| font-weight: lighter; | Тонкий шрифт; |
| font-size: 16px; | Размер шрифта; |
| font-family: Georgia | Название шрифта; |
| font-variant: small-caps; | Делает все буквы прописными; |
| font:italic bold 16px Georgia; | Сокращение записи для свойства font |

### Внешние и внутренние отступы

| margin-top: 20px; | Внешний отступ сверху; |
| --- | --- |
| margin-right: 20px; | Внешний отступ справа; |
| margin-bottom: 20px; | Внешний отступ снизу; |
| margin-left: 20px; | Внешний отступ слева; |
| margin: 20px 20px 20px 20px; | Сокращённая запись для внешних отступов, первая цифра отступ сверху, и далее по часовой стрелке; |
| padding-top: 10px; | Внутренний отступ сверху; |
| padding-right:10px; | Внутренний отступ справа; |
| padding-bottom: 10px; | Внутренний отступ снизу; |
| padding-left: 10px; | Внутренний отступ слева; |
| padding: 10px 10px 10px 10px; | Сокращение записи для свойства padding, расположение аналогично margin |

### Свойства фона

| background-color: #00ff00; | Цвет фона элемента; |
| --- | --- |
| background-position: center; | Размещает фоновое изображение по центру; |
| background-position: 100px 150px; | Размещает фоновое изображение, первая цифра — отступ сверху (top), вторая — отступ слева left; |
| background-repeat:repeat; | Изображение повторяется пока не займёт всё пространство блока; |
| background-repeat: no-repeat | Изображение не повторяется; |
| background-repeat:repeat-y | Изображение повторяется по вертикали; |
| background-repeat:repeat-x; | Изображение повторяется по горизонтали; |
| background-attachment: fixed; | Изображение фиксируется и остаётся на месте при прокрутке страницы; |
| background-image:url(images/im.png) | Вставляет изображение в элемент; |
| background: #00ff00 url(images/im.png) no-repeat fixed center; | Сокращение записи для background; |
| background-size: | Масштабирует фоновое изображение; |

### Ширина и высота

| width: 100px; | Ширина элемента; |
| --- | --- |
| min-width: 100px; | Минимальная ширина элемента; |
| max-width: 100px; | Максимальная ширина элемента; |
| height: 100px; | Высота элемента; |
| min-height: 100px; | Минимальная высота элемента; |
| max-height: 100px; | Максимальная высота элемента; |

### Рамки

| border: 3px; | Рамка элемента; |
| --- | --- |
| border-color:#ff0000; | Цвет рамки; |
| border-style: solid; | Стиль рамки; |
| border: 5px solid #ff0000; | Сокращение записи для рамки; |
| outline: 5px solid #00FF00; | Задаёт рамку за внешней границей элемента; |
| border-radius: #00FF00 25px; | Создаёт рамку с закруглёнными углами вокруг элемента; |

### Свойства списка

| list-style-type: | Определяет вид маркера в списке. Применяемые значения:  none — без маркера;  circle — круг;  disc — заполненный круг;  square — квадрат;  upper-alpha — A B C D …;  upper-roman — I II III IV V …;  lower-alpha — a b c d …;  decimal-leading-zero — 01 02 03 04 …;  none — отменяет маркер |
| --- | --- |
| list-style-image: url(images/im.png) | Устанавливает маркер-изображение; |
| list-style-position: inside; | Размещает маркер-изображение внутри блока списка; |
| list-style:square inside url(images/im.png); | Сокращение записи для list-style, square указывается на случай недоступности картинки; |

### Позиционирование

| position: absolute; | Размещает элемент относительно границ экрана монитора. Применяется со свойствами: top:, left:, right:, bottom:, в которых указываются координаты позиционирования |
| --- | --- |
| position: relative; | Применяется к элементу, относительно которого позиционируется другой элемент; |
| position: fixed; | Фиксирует элемент в указанной части экрана. Элемент не реагирует на прокрутку |
| x-index: ; | Накладывает элементы друг на друга по оси х; |
| y-index: ; | Накладывает элементы друг на друга по оси у; |
| z-index: ; | Накладывает элементы друг на друга по оси z; |
| float: right; | Задаёт обтекание слева; |
| float: left; | Задаёт обтекание справа; |
| clear: both; | Запрещает обтекание; |
| clip: rect(10px,50px); | Обрезает изображение квадратом; |

### Видимость

| visibility: hidden; | Делает элемент невидимым, но сохраняет занимаемое им место; |
| --- | --- |
| display: none; | Временно удаляет элемент со страницы, и другие элементы могут занять его место; |
| display: block; | Делает элемент блочным; |
| display: inline; | Делает элемент встроенным; |
| display: inline-block; | Делает элемент встроенным, но с ддоступными свойствами блочного; |
| overflow: auto; | При необходимости добавляет полосу прокрутки в блок с контентом; |
| opacity: 0; | Делает элемент полностью прозрачным. Прозрачность уменьшается до значения 1; |

### Трансформация и анимация

| transform | поворачивает, сдвигает, наклоняет и масштабирует элемент. |
| --- | --- |
| transform: scale(); | Масштабирует элемент по горизонтали и вертикали. Число в скобках больше единицы увеличивает; элемент, а меньше единицы — уменьшает |
| transform: scaleX(); | Масштабирует элемент по горизонтали; |
| transform: scaleY(); | Ьасштабирует элемент по вертикали |
| transform: skewX(); | Наклоняет элемент на заданный угол в скобках по вертикали влево, отрицательное значение — вправо |
| transform: skewY(); | Наклоняет элемент на заданный угол по горизонтали; |
| transform: translateX(); | Сдвигает элемент по горизонтали |
| transform: translateY(); | Сдвигает элемент по вертикали |
| transform: rotate(); | Поворот элемента вокруг центра на заданный угол |
| transform: rotateY(360deg); | Поворот элемента вокруг вертикальной оси. Применяется в 3D |
| transform: rotateX(360deg); | Поворот элемента вокруг горизонтальной оси. Применяется в 3D |
| transform: matrix() | Создаёт эффект отражения элемента |
| transform-style: preserve-3d | Создаёт 3D эффект |
| transform-origin: x y z; | Задаёт координаты точки относительно центра, вокруг которой будет вращаться элемент |
| transition-property: | Определяет свойство, которое будет анимироваться |
| transition-duration: | Продолжительность эффекта перехода |
| transition-timing-function: | Скорость эффекта перехода |
| transition-delay: | Задержка начала анимации |
| transition: left 3s ease; | Сокращение записи для transition |
| animation | Pадаёт параметры действия для анимации — длительность анимации, повторение, направление, тип движения, шаги. Работает со свойством @keyframes |
| animation-name | Имя анимации |
| animation-duration | Длительность анимации |
| animation-timing-function | Определяет вид анимации |
| animation-iteration-count | Задаёт число повторений анимации |
| Animation-direction | Задаёт направление движения |
| animation-delay | Задаёт задержку анимации перед началом воспроизведения |
| animation: nev 5s infinite alternate; | Сокращение записи для animation |
| @keyframes | Правило, записывается как селектор, внутрь которого помещаются селекторы последовательно определяющие состояние элемента в определённый момент |

### Добавление элементов без html

| content: | Выводит на страницу элементы, которых нет в html коде. Применяется с псевдоклассами ::after и ::before |
| --- | --- |
| content:"" | В кавычки можно добавить текст и он будет выведен на страницу |
| content:attr(); | В скобки обычно добавляется спецсимвол html, изображение которого и выводится в начале или в конце элемента |
| content: url(images/1.png); | Выводит картинку в начале или в конце элемента. Обычно маркер списка. |

### Колончатая вёрстка страницы

| columns: | Разделяет поле страницы на колонки |
| --- | --- |
| column-count: 3; | Количество колонок |
| column-width: 200px; | Ширина колонок |
| column-gap: 1-px; | Ширина пробела между колонками |
| column-span: all; | Указывает количество колонок, которые пересечёт элемент. Применяется обычно для заголовка. |
| column-rule-style: solid; | Создаёт вертикальную разделительную линию между колонками. |
| column-rule-width: 2px; | Ширина разделительной линии |
| column-rule-color: #999; | Цвет разделительной линии |
| column-rule: 2px solid #999; | Краткая запись разделительной линии |

### Медиа запросы

| @media all{Таблица стилей} | Селекторы и свойства css заносятся в фигурные скобки и применяется ко всем типам носителей |
| --- | --- |
| @media screen and (){} | Экран монитора. В круглых скобках указываются дополнительные требования, например размер экрана |
| @media tv and (){} | Телевизор |
| @media print and (){} | Принтеры |
| @media handheld and (){} | Смартфоны |
| @media screen, handheld and (){} | Монитор и смартфон |
| @media screen and (max-width: 600px){} | Для моноторов с разрешением менее 600px |

### Другие @ правила

| @charset «кодировка»; | Устанавливает кодировку для css файла |
| --- | --- |
| @import «имя файла» [типы носителей]; | Импортирует содержимое CSS-файла в текущую стилевую таблицу |

### Цветовые и визуальные фильтры

| filter: blur(3px); | Размытие |
| --- | --- |
| filter: brightness(70%); | Яркость |
| filter: contrast(50%); | Контрастность |
| filter: drop-shadow(2px 3px 5px blue); | Добавляет тень ко всем элементам |
| filter: grayscale(50%); | Обесцвечивание |
| filter: hue-rotate(70deg); | Изменяет цвета в зависимости от заданного угла в цветовом круге |
| filter: invert(100%); | Негатив изображения |
| filter: opacity(30%); | Прозрачность |
| filter: saturate(300%); | Насыщенность цвета. Меньше 100% — уменьшает насыщенность, больше 100% — увеличивает |
| filter: sepia(120%); | Ретро |

### Flexbox

| .flex-container{} | Преобразует любой элемент во флекс-контейнер, а все дочерние элементы во флекс-элементы. Обязательное свойство display: flex; |
| --- | --- |
| justify-content: flex-start; | Дочерние элементы позиционируются от начала контейнера |
| justify-content: flex-end; | Дочерние элементы позиционируются от конца контейнера (справа) |
| justify-content: center; | Дочерние элементы позиционируются по центру контейнера |
| justify-content: space-between; | Элементы выравниваются по главной оси. Первый элемент располагается в начале контейнера, последний в конце, а все остальные равномерно располагаются на свободном пространстве между ними |
| justify-content: space-around; | Свободное пространство делится поровну между элементами, добавляя им отступы |
| justify-content: inherit; | Наследует родительское свойство |
| align-items: stretch; | Флекс элементы занимают всё пространство по высоте. |
| align-items: flex-start; | Все дочерние элементы выравниваются по левому краю и верхней границе flex-бокса |
| align-items: flex-end; | Элементы выравниваются по нижнему краю, относительно нижней границы flex-бокса |
| align-items: center; | Выравнивание по центру |
| align-items: baseline; | Выравнивание по базовой линии |
| align-items: inherit; | Наследует значение родителя |
| flex-direction: row; | Направление главной оси слева направо |
| flex-direction: row-reverse; | Направление главной оси справа налево |
| flex-direction: column; | Направление главной оси сверху вниз |
| flex-direction: column-reverse; | Направление снизу вверх |
| flex-wrap: nowrap; | Flex-элементы не переносятся и располагаются в одну строку слева направо |
| flex-wrap: wrap; | Элементы переносятся и располагаются в несколько рядов слева на право |
| flex-wrap: wrap-reverse; | Элементы переносятся и располагаются в обратном порядке. Перенос осуществляется снизу вверх |
| align-content: stretch; | Строки равномерно растягиваются, заполняя всё доступное пространство |
| align-content: flex-start; | Строки выравниваются по левому краю,относительно верхней границы |
| align-content: flex-end; | Строки выравниваются по левому краю, относительно нижней границы |
| align-content: center; | Выравнивание по центру |
| align-content: space-between; | Строки выравниваются по высоте, свободное пространство распределяется между ними |
| align-content: inherit; | Наследует значение родителя |
| order: -1 или 1; | Ставит любой flex-элемент в начало строки или в конец строки |
| flex-basis: 200px; | Ширина элемента задаётся во всех доступных в веб размерах |
| flex-basis: auto; | Элемент получает базовую ширину, соответствующую ширине контента |
| flex-grow: 1.0 ….; | Устанавливает коэффициент увеличения элемента |
| flex-shrink: 1.0 …; | Устанавливает коэффициент уменьшения элемента |
| align-self: auto; | Выравнивание отдельного элемента. По умолчанию задаётся выравнивание родителя |
| align-self: flex-start; | Отдельный элемент выравнивается по левому краю, относительно верхней границы |
| align-self: flex-end; | Выравнивается по левому краю, относительно нижней границы |
| align-self: center; | Выравнивается по центру |
| align-self: baseline; | Выравнивается по базовой линии слева |
| align-self: stretch; | Элемент растягивается на всю высоту flex-контейнера |