

# Вложенные элементы

Неупорядоченные списки

Упорядоченные списки

Списки описаний

# Неупорядоченные списки

Элемент `<ul>` является родительским элементом для неупорядоченных списков элементов. Единственными дочерними элементами `<ul>` являются один или несколько элементов списка `<li>`

```
<ul>  
  <li>Blender</li>  
  <li>Toaster</li>  
  <li>Vacuum</li>  
</ul>
```

# Упорядоченные списки

Элемент `<ol>` является родительским элементом для упорядоченных списков элементов. Единственными дочерними элементами `<ol>` являются один или несколько элементов `<li>`

атрибут `type` определяет тип нумерации

<pre>&lt;ol&gt;  &lt;li&gt;Blender&lt;/li&gt;  &lt;li&gt;Toaster&lt;/li&gt;  &lt;li&gt;Vacuum&lt;/li&gt;  &lt;/ol&gt;</pre>	<pre>&lt;ol type="A"&gt;  &lt;li&gt;Blender&lt;/li&gt;  &lt;li&gt;Toaster&lt;/li&gt;  &lt;li&gt;Vacuum&lt;/li&gt;  &lt;/ol&gt;</pre>
---	--

Атрибут `boolean reversed`, если он включен, изменит порядок следования чисел, начиная с наибольшего числа и заканчивая наименьшим. Атрибут `start` задает начальное значение.

```
<ol reversed start="6">
```

```
<li>Blender</li>
```

```
<li>Toaster</li>
```

```
<li>Vacuum</li>
```

```
</ol>
```

атрибут value задает номер <li>

```
<ol start="4">
```

```
<li value="7">Blender</li>
```

```
<li value="29">Toaster</li>
```

```
<li>Vacuum</li>
```

```
</ol>
```

```
<ol type="A" start="33">
```

```
<li value="7">Blender</li>
```

```
<li value="29">Toaster</li>
```

```
<li>Vacuum</li>
```

```
</ol>
```

# Списки описаний

Список описаний — это элемент `description list` (`<dl>`), содержащий ряд (ноль или более `description terms` (`<dt>`) и их `description details` (`<dd>`).

```
<dl>
  <dt>Blendan Smooth</dt>
  <dd>Originally a margarita maker, they are now an aspiring load balancer.</dd>
  <dt>Toasty McToastface</dt>
  <dd>Formerly partially to fully baked, they are now an aspiring nuclear codes handler.</dd>
</dl>
```

# Многоуровневый список в HTML

```
<ol>  
  <li>18 июля  
    <ol>  
      <li>Написать статью</li>  
      <li>Почитать книгу 1 час</li>  
      <li>Сходить в кино</li>  
    </ol>  
  </li>  
  <li>19 июля</li>  
  <li>20 июля</li>  
</ol>
```

# Зачем нам необходима структура?

Пользователи, просматривающие веб-страницу, быстро сканируют её в поиске подходящего контента, часто просто просматривая только заголовки .

Поисковые системы, индексирующие вашу страницу, считают содержание заголовков важными ключевыми словами для влияния на ранжирование поиска страницы. Без заголовков ваша страница будет плохо работать с точки зрения [SEO](#) (Search Engine Optimization — поисковая оптимизация).

Сильно слабовидящие люди часто не читают веб-страницы — они слушают их вместо этого. Это делается с помощью программного обеспечения, называемого [программой чтения с экрана](#). Это программное обеспечение предоставляет способы быстрого доступа к данному текстовому контенту. Среди различных используемых методов они предоставляют схему документа, считывая заголовки, позволяя своим пользователям быстро находить нужную им информацию. Если заголовки недоступны, они будут вынуждены слушать весь документ вслух.

Чтобы стилизовать контент с помощью [CSS](#) или сделать его интересным с помощью [JavaScript](#), вам нужно, чтобы элементы обёртывали соответствующий контент, чтобы CSS и JavaScript смогли эффективно работать.



# Основы редактирования текста в HTML

## основные элементы: <p> ; <h1>-<h6>; <span>

<h1>Сокрушительная скука</h1>

<p>Крис Миллс</p>

<h2>Глава 1: Тёмная ночь</h2>

<p>Это была тёмная ночь. Где-то кричала сова. Дождь обрушился на ...</p>

<h2>Глава 2: Вечное молчание</h2>

<p>Наш главный герой ничего не мог, когда шёпот из тёмной фигуры ...</p>

<h3>Призрак говорит</h3>

## Акцент и важность

<p>Я **<em>рад</em>**, что ты не **<em>опоздал</em>**.</p>

<p>Эта жидкость **<strong>очень токсична</strong>**.</p>

**<i> (en-US)** используется для передачи значения, традиционно передаваемого курсивом: иностранные слова, таксономические обозначения, технические термины, мысли ...

**<b>** используется для передачи значения, традиционно передаваемого жирным шрифтом: ключевые слова, названия продуктов, предложения ...

**<u> (en-US)** используется для передачи значения, традиционно передаваемого подчёркиванием: имя собственное, орфографическая ошибка ...

<p>

Колибри обыкновенный (<i>архилоус обыкновенный</i>) — наиболее часто встречающийся вид колибри в северо-восточной Америке.

</p>

<p>

Случился прилив иностранных слов, таких как <i lang="uk-latn">vatrushka</i> ,

<i lang="id">nasi goreng</i> и <i lang="fr">soupe à l'oignon</i> .

</p>

<p>Когда-нибудь я узнаю, как <u>гаварить</u> без ошибок.</p>

# Таблицы

## 1. структура таблиц:

**<table>** представляет табличные данные, то есть информацию, представленную в двумерной таблице, состоящей из строк и столбцов ячеек, содержащих данные.

**<thead>** инкапсулирует набор строк таблицы (элементов [<tr>](#)), указывая, что они составляют **заголовок таблицы** со сведениями о столбцах таблицы.

**<tbody>** инкапсулирует набор строк таблицы указывая, что они **составляют тело** (основных) данных таблицы.

**<tfoot>** инкапсулирует набор строк таблицы (элементов [<tr>](#)), указывая, что они составляют **нижнюю часть** таблицы со сведениями о столбцах таблицы.

**<tr>** определяет строку ячеек в таблице.

**<th>** определяет ячейку как заголовок группы ячеек таблицы и может использоваться как дочерний элемент

**<td>** определяет ячейку таблицы, содержащую данные и используемую в качестве дочерней ячейки элемента [<tr>](#).

```
<table>
  <thead>
    <tr>
      <th></th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td></td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td></td>
    </tr>
  </tfoot>
</table>
```

```
<table>
  <thead>
    <tr>
      <th>название </th>
      <th>количество </th>
      <th>цена за литр </th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>молоко </th>
      <td>2 </td>
      <td>10 </td>
    </tr>
    <tr>
      <th>хлеб </th>
      <td>1 </td>
      <td>4 </td>
    </tr>
    <tr>
      <th>чай </th>
      <td>3 </td>
      <td>20 </td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td colspan="2" class="total">Итого: </td>
      <td>84 </td>
    </tr>
  </tfoot>
</table>
```

## 2. Атрибуты таблиц

Атрибут `colspan` задает количество ячеек, объединенных по горизонтали,

а `rowspan` — по вертикали.

Можно использовать `border`. (но лучше его задавать в `css`)

# картинка

1) ``  
атрибуты

`src` — базовый и обязательный, включающий в себя ссылку на изображение. Она может быть прописана как до файла, размещенного на другом веб-ресурсе, так и до изображения, которое находится на домашнем компьютере.

`height` — позволяет изменять высоту изображения, пропорционально которому изменяется ширина.

`width` — позволяет изменять ширину изображения, пропорционально которой изменяется высота.

Изображение как ссылка на другую страницу:

`<a href="https://www.google.ru/" target="blank">`  
``

`</a>`



2\*) <style>

body{

background-image:

url('https://na-dache.pro/uploads/posts/2021-05/1621545702\_37-p-peizazhi-prirodi-45.jpg');

}

</style>

`background-repeat: no-repeat;` — отображается только один раз без

повторения.

`background-size: cover;` — масштабируется так, чтобы полностью заполнить фон элемента, возможно, обрезая его.

`background-size: contain;` — масштабируется так, чтобы полностью поместиться в фон элемента без искажений. Белые полосы могут появиться по краям фона, если пропорции изображения и фона не совпадают.

# CSS

CSS (Cascading Style Sheets) - это язык стилей, который используется для описания внешнего вида документа, написанного с использованием языка разметки, такого как HTML или XML. Он позволяет разделить содержимое документа от его внешнего вида, что позволяет легко изменять внешний вид сайта без изменения самого содержимого.

В CSS есть три основные сущности: селекторы, правила и значения.

-Селекторы используются для выбора элементов на странице, которые нужно изменить. Существует множество различных типов селекторов, таких как теги, классы и идентификаторы.

Классы и идентификаторы должны прописываться в HTML документе для какого-нибудь элемента, который вы хотите изменить отдельно от других. Классы прописываются с помощью атрибута `class` со значением равным названию класса. В CSS элементы с классом указываются с помощью точки после которой идет название класса.

Идентификаторы аналогично, с помощью атрибута `id` и в CSS указываются с помощью `#`.

-Правила - это инструкции, которые определяют, как должен выглядеть выбранный элемент. Правило состоит из свойства и значения.

-Значения - это то, что присваивает свойству определенное значение. Например, если свойство - это цвет текста, то значение может быть "красный" или "синий".

```
селектор
body {
  свойства значения
width: 100px;
свойства значения
color: red;
}
```

# Единицы измерения

Пиксель **px** – это самая базовая, абсолютная и окончательная единица измерения.

Количество пикселей задаётся в настройках [разрешения экрана](#), один **px** – это как раз один такой пиксель на экране. Все значения браузер в итоге пересчитает в пиксели.

Пиксели могут быть дробными, например размер можно задать в 16.5px.

**Достоинства** : Главное достоинство пикселя – четкость и понятность

**Недостатки**: Другие единицы измерения – в некотором смысле «мощнее», они являются относительными и позволяют устанавливать соотношения между различными размерами

1 **em** – текущий размер шрифта.

Можно брать любые пропорции от текущего шрифта: 2em, 0.5em и т.п.

Размеры в em – *относительные*, они определяются по текущему контексту.

Проценты %, как и em – относительные единицы.

Когда мы говорим «процент», то возникает вопрос – «Процент от чего?»

Как правило, процент будет от значения свойства родителя с тем же названием, но не всегда.

Это очень важная особенность процентов, про которую, увы, часто забывают.

Итак, мы рассмотрели:

px – абсолютные, чёткие, понятные, не зависящие ни от чего.

em – относительно размера шрифта.

% – относительно такого же свойства родителя (а может и не родителя, а может и не такого же – см. примеры выше).

Единица **rem** задаёт размер относительно размера шрифта элемента <html>.

Во всех современных браузерах, исключая IE8-, поддерживаются новые единицы из черновика стандарта [CSS Values and Units 3](#):

**vw** – 1% ширины окна

**vh** – 1% высоты окна

**min** – наименьшее из (vw, vh), в IE9 обозначается **min**

**max** – наибольшее из (vw, vh)

Эти значения были созданы, в первую очередь, для поддержки мобильных устройств.

Их основное преимущество – в том, что любые размеры, которые в них заданы, автоматически масштабируются при изменении размеров окна.