

## Практическая работа №2

В прошлой работе мы спроектировали структуру сайта, изучили как размечать крупные структурные блоки и немного коснулись разметки текста. В этой работе будем работать с содержимым нашего сайта и изучим, как ещё можно размечать разные текстовые элементы. Расширять список известных тегов начнем с тегов dl, dt и dd.

Данные теги являются списком описаний. Список описаний хорош для разметки вопросов-ответов, наименований и определений, категорий и тем. Он создаётся с помощью трёх тегов:

- dl - (сокращение от «description list») обозначает сам список описаний;
- dt - (сокращение от «description term») обозначает термин;
- dd - (сокращение от «description definition») обозначает описание или определение.

Теги dt и dd пишутся внутри dl. Каждый список может содержать один или несколько терминов и одно или несколько описаний для каждого термина. Пример:

```
<dl>
  <dt>HTML</dt>
  <dd>Язык гипертекстовой разметки</dd>
</dl>
```

Задание

1. Откройте созданный в первой работе index.html в редакторе кода
2. Внутри <main> создайте дополнительный раздел <section> ниже всех остальных
3. Внутри него добавьте список описаний для термина “Жажда”, определением термина будет “физиологическое ощущение, относящееся к разряду общих чувств и служащее сигналом того, что организм нуждается в воде. Решением является употребление ...”

## Теги sub и sup, нижний и верхний индексы

Запись о решении проблемы жажды оказалась неполной. Всем понятно что в случае с жаждой необходимо выпить простой воды, но для нас - необходимым станет написать  $H_2O$ , для того чтобы изучить новые теги. Тег `<sup>` (сокращение от

superscript) - отображает текст в виде верхнего индекса, а тег `<sub>` (сокращение от subscript) - в виде нижнего индекса. Эти теги чаще всего используются для выделения отдельных символов, а не слов. Их используют для указания единиц измерения или написания простых формул. Например для написания сокращенных единиц измерения, таких как квадратные метры используется запись:

`м<sup>2</sup>`

`<sup>` выносит вложенный в него символ на полстроки выше. Соответственно `<sub>` делает обратное

### Задание

1. Вместо троеточия вставьте формулу воды

## Тег pre, преформатированный текст

В этой записи мы экспериментируем с ASCII-графикой и попытаемся нарисовать котика. Заранее скажу что ничего не получится, так как отображение кода в редакторе и в браузере отличаются. Вы можете ставить сколько угодно пробелов в HTML-коде, но браузер всё равно отобразит их как один. То же самое с переносами строк — они не сохраняются при отображении в браузере. Изменить это поведение можно с помощью тега `<pre>` (сокращение от «preformatted text»), который обозначает «предварительно отформатированный текст». Чаще всего этот тег используется для отображения примеров кода, реже применяется для отображения картинок ASCII Art. Браузер сохраняет и отображает все пробелы и переносы, которые есть внутри тега

## Задание

1. Создайте новый параграф ниже, а внутри него

```

-
\'*
) -
:
:
:
-
\'
:
*
:
*
;
-
\'
;
;
:
:
:
\'
:
:
+
;
:
:
:
|
;
;
:
:
:
:
*
:
*
/
*
*
*
+
*
*
*
*
*
*

```

Форматирование должно остаться четко таким

2. Измените тег параграфа на тег <pre>, сохраните и обновите страницу в браузере

## Символы-мнемоники, тег code, комбинирование

В новом тексте вам захотелось перечислить теги. И чтобы они обязательно выглядели в браузере так же, как в коде. Например, `<h1>` или `<br>`. Задача это непростая. Некоторые символы в HTML зарезервированы, то есть браузер считает их HTML-кодом. Например, любой текст после знака меньше (`<`) браузер будет пытаться интерпретировать как тег и на странице не отобразит. Каким образом специальные символы (например больше и меньше) можно использовать в тексте страницы как обычные символы? Для этого их нужно заменить на символы-мнемоники. Мнемоники — это особые строки, которые начинаются с амперсанда (&) и заканчиваются точкой с запятой (;). Например, знак меньше на страницу можно вставить мнемоникой `&lt;` (less than), а знак больше мнемоникой `&gt;` (greater than):

`&lt;h2&gt;` Заголовок страницы `&lt;/h2&gt;`

Существует множество других символов-мнемоник. Например: `&copy;` - ©, `&laquo;` - «, `&raquo;` - », `&sect;` - §. Наиболее полный список мнемоник представлен в справочнике <https://html.spec.whatwg.org/multipage/named-characters.html>

Теги в тексте становятся похожи на настоящие теги. Но пока что это просто текст, обычные слова. Давайте выделим их как фрагменты кода. Для обозначения фрагментов кода есть специальный тег, который так и называется `<code>`. С его помощью размечается любой фрагмент текста, который распознается компьютером: код программы, разметки, название файла и так далее. Какого-то специального способа указать, на каком языке написан фрагмент кода, у тега нет. Обычно браузеры отображают текст в теге моноширинным шрифтом. Пример разметки с `<code>`:

`<p>Тег <code>&lt;ul&gt;</code> — это неупорядоченный список.</p>`

Допустим нам необходимо дополнить текст записи наглядным примером: многострочным фрагментом кода с хорошим форматированием. Как вы помните, сохранить форматирование помогает тег `<pre>`, а для разметки кода используется тег `<code>`. Получается, эти теги нужно скомбинировать, то есть вложить один тег в другой.

Главное — соблюдать правильный порядок вложенности. Тег `<code>` можно вкладывать внутрь тега `<pre>`, а наоборот делать нельзя. Пример:

```
<pre>
  <code>
    if (a > b) {                // Некий пример
      console.log('Hello!');    // абстрактного JS кода
    }
  </code>
</pre>
```

Никакой разницы в отображении просто тега `<pre>` и тега `<pre>` с вложенным `<code>` в браузере не будет, но второй вариант разметки информативнее и выразительнее.

## Задание

1. Вставьте данные примеры в нашу секцию

## Тег q, cite, blockquote

В HTML существует несколько тегов для разметки цитат. Самый простой — тег `<q>` (сокращение от «quote»). Он предназначен для выделения цитат внутри предложения. Текст внутри этого тега браузер автоматически обрамляет кавычками, поэтому добавлять кавычки вручную не нужно. Если известен интернет-адрес цитаты, то его можно указать в атрибуте `cite`:

`<p>В спецификации говорится: <q cite="https://www.w3.org/TR/html52/">The q element represents some phrasing content quoted from another source.</q></p>`

Но не путайте атрибут `cite` с одноименным тегом `<cite>`, который тоже относится к цитатам. Тег `<cite>` — «расширенная версия» атрибута `cite`. В нём можно указывать помимо адреса источника цитаты ещё и название произведения, откуда цитируется текст, а также имя автора или организации, чей текст цитируется. Содержимое в браузере выделяется курсивом. Когда теги `<q>` и `<cite>` используют вместе, их ставят рядом друг с другом:

`<p> И этими словами заканчивалась сцена<cite>Гендальфа Белого</cite> — <q>Ни силы в тебе нет, ни мудрости. Однако же избран ты, а значит, придётся тебе стать сильным, мудрым и доблестным.</q></p>`

Но представим, что наша цитата объемнее пары фраз. Для выделения длинных цитат, которые могут состоять из нескольких абзацев, предназначен тег `<blockquote>`. Он выделяет цитату не как фрагмент текста в предложении, а как отдельный блок текста с отступами. Так же, как и у `<q>`, у `<blockquote>` может быть атрибут `cite`, содержащий ссылку на источник цитаты. И также в паре с `<blockquote>` можно использовать тег `<cite>`, который обычно размещается внутри цитаты. Пример:

```
<blockquote>
  <p>Если тебе где-то не рады в рваных носках, то и в целых туда идти не
стоит</p>
  <cite>Джейсон Стэтхэм</cite>
</blockquote>
```

## Задание

1. Добавьте на свою страницу пару цитат, разными способами

## Акцентирование внимания, важность и выделение

Со временем вы замечаете, что хотите выделить некоторые фрагменты текста ваших записей. И не просто выделить, а сделать на них особый акцент. Следующие два тега предназначены для акцентирования внимания на слово или фразу. Это теги `<em>` (сокращение от «emphasis») и `<i>` (сокращение от «italic»). Визуально оба тега одинаковы, они выделяют текст курсивом. Но значение у этих тегов различное. Тег `<em>` определяет текст, на который сделан особый акцент, меняющий смысл предложения. Например, если автор текста любит лето и недолюбливает зимнее время, то его речь можно разметить следующим образом:

`<p>Я просто обожаю холодные зимние дни!  
</p>`

Тег `<i>` применяется для обозначения текста, который отличается от окружающего текста, но не является более важным. Например, в `<i>` можно заключать названия, термины, иностранные слова. Также в этот тег можно обернуть мысли героя. В речи такой текст обычно выделяется интонационно. Приведём пример:

`<p>Он взглянул в окно и подумал — такого просто не может быть!  
</p>`

На этом возможности выделения слов и фраз в тексте не заканчиваются. Вы научились выделять фразы курсивом, а теперь давайте разберёмся с выделением текста жирным. Теги `<strong>` и `<b>` (сокращение от «bold») предназначены для выделения слова или фразы. Отображаются оба тега одинаково, они выделяют текст жирным. Тег `<strong>` указывает на важность отмеченного текста. Он может использоваться для выделения предупреждений или части документа, которую пользователь должен увидеть раньше остального. При этом обозначение части текста тегом `<strong>` не должно изменять смысла предложения.

`<p><strong>Внимание!  
</strong> Это место опасно. <strong>Вы можете упасть в пропасть,  
</strong> если подойдёте близко к краю.  
</p>`

Тег `<b>` предназначен для выделения текста с целью привлечения к нему внимания, но без придания ему особой важности. Использовать его нужно только в случае, когда остальные теги выделения не подходят. Типичный пример — выделение вводного предложения статьи.

Фродо вошел в небольшую комнату. Его **меч** загорелся ярче, а это означало только одно - **орки**!

Лучше всего отличия этих тегов будут заметны людям, которые используют средства доступности, в частности, слепым и слабовидящим. Скринридер при чтении сайта будет выделять слова с тегом `<strong>` интонационно, в отличие от простого выделения с помощью `<b>`. То же самое касается тегов `<em>` и `<i>`. Тег `<em>` «читалка» будет выделять интонацией.

## Теги del и ins, выделение изменений

Списки дел на то и нужны, чтобы вычеркивать выполненные дела и иногда добавлять новые. У вас также назрела необходимость обновить список дел, но так, чтобы было видно, какие дела завершены, а какие добавлены. Как сделать разметку такой истории изменений? Как раз для описания изменений предназначены теги `<del>` (сокращение от «delete») и `<ins>` (сокращение от «insert»). Тег `<del>` выделяет текст, который был удален в новой версии документа. В браузере этот текст перечеркивается. Тег `<ins>` выделяет текст, который был добавлен в новой версии документа. В браузере этот текст подчеркивается. Оба тега имеют атрибут `datetime`, в котором можно указать дату и время, когда была внесена та или иная правка. Простейшим примером применения этих тегов может служить список дел. Когда дело выполнено, его помечают тегом `<del>`, а если появилось новое дело, то его добавляют в список и помечают тегом `<ins>`.

```
<ul>
  <li>Выгулять пса</li>
  <li>
    <del datetime="2021-08-11T01:25-07:00">Дочитать книгу</del>
  </li>
  <li>
    <del datetime="2021-08-10T23:38-07:00">Принять душ</del>
  </li>
  <li>
    <ins>Купить лейз со вкусом майонеза</ins>
  </li>
</ul>
```

Атрибут `datetime` предназначен не для людей, а для компьютеров, поэтому дату и время там пишут в стандартизованном формате. При такой разметке программам легче разбирать документы и анализировать, когда произошли те или иные изменения.



## Тег time, дата и время

В новой записи вы решили уточнить точное время дня (а точнее ночи), когда именно была сделана запись. Строку со временем можно оставить просто текстом, но есть вариант получше.

В HTML есть специальный тег для разметки даты и времени. Это тег `<time>`.

С помощью `<time>` можно описывать даты одновременно и для человека, и для машины. Дата для человека описывается внутри тега, а дата для машины — внутри атрибута `datetime` в формате ISO 8601:

```
<time datetime="2016-11-18T09:54">09:54 утра</time>  
<time datetime="2015-11-18">18 ноября 2015</time>  
<time datetime="2018-09-23">в прошлую субботу</time>  
<time datetime="2017-04-20">вчера</time>
```

Браузер отображает только содержимое тега, а содержимое `datetime` не отображается. Человек увидит только понятное ему обозначение времени, а машина прочитает атрибут и получит дату в нужном ей формате. И все останутся довольны.

## Тег br, перенос строки

Давайте попробуем написать стихотворение. Как мы выяснили на примере рисования ASCII-котэ - нам будет не хватать переносов строк внутри строфы. Мы уже знаем много тегов, может какие-то из них подойдут для разбиения текста на строки? Первый кандидат — тег `<r>`. Идеально подходит для выделения строфы, а нам нужно добавить переносы внутрь строф. Отмечаем вариант. Второй кандидат — тег `<pre>`. Сохраняет всё форматирование текста. Вроде подходит, но нам нужны только переносы строк. В общем, ничего не подходит.

Нужен специальный тег, который просто добавляет перенос строки внутри абзаца. Для этого в HTML предусмотрен одиночный тег `<br>` (сокращение от «line break»). Этот тег отлично подходит, если переносы нужны в тексте для повышения его читабельности, например, в почтовых адресах, стихах, текстах песен, режиме работы.

Иногда этот тег применяют неправильно, для разбиения текста на «как бы абзацы». Никогда так не делайте. Для разметки абзацев нужно использовать тег `<p>`

## Теги div и span, контейнеры для стилизации

Вы уже узнали так много новых тегов и у каждого из них есть свой смысл. Но иногда хочется просто красиво, без придания какого-то смысла, выделить короткий фрагмент текста или несколько абзацев. Например, подсветить самые опасные на ваш взгляд болезни верстальщика красноватым фоном.

Но раз такое выделение «бессмысленно», то и делать его «смысловыми» тегами нельзя? Да, это так. Но есть два специальных тега у которых «смысла» нет. Это теги `<div>` (сокращение от «division») и `<span>`. Это «чистые» элементы, которые отлично подходят для визуальной группировки других элементов. Использовать эти теги рекомендуется, если более подходящих семантических тегов не нашлось.

Теги `<div>` и `<span>` не имеют никакого оформления по умолчанию и их почти всегда используют вместе с атрибутом `class`, чтобы легко добавлять им собственные стили.

Тег `<div>` обычно используется для группировки крупных элементов, например, нескольких абзацев, или в качестве контейнера для создания сеток страниц. А `<span>` используется для выделения мелких текстовых элементов: частей слов, отдельных слов или фраз, состоящих из нескольких слов:

```
<article>
  <div class="highlight">
    <p>Первая строка ничего не значащего текста.</p>
    <p>А тут вторая</p>
  </div>
  <p>Текст, в котором <span>выделена фраза</span>.</p>
</article>
```

В этом задании мы используем `<div>` в качестве обёртки для пунктов списка определений. В таком случае в теге `<dl>` не должно находиться не обёрнутых в `<div>` пунктов списка. Мы либо оборачиваем все элементы, либо оставляем все теги `<dt>` и `<dd>` без обёрток. Смешение обёрнутых и не обёрнутых в `<div>` определений внутри одного списка считается ошибкой.

На самом деле `<div>` - самый часто используемый тег в построении веб-страниц. Элемент разделения контента `<div>` является универсальным контейнером для потокового контента. Он не влияет на контент или макет до тех пор, пока не будет стилизован с помощью CSS. Его атрибут `class` в примере выше - необходим как раз для дальнейшей стилизации блока.

### Задание

1. Создайте несколько акцентов на своей странице
2. Используйте теги `del` и `ins` для “утраты” одного из навыков и “приобретения” нового, в нем же используйте тег `time` для указания времени
3. Создайте связь контента с помощью тегов `div` и `span`
4. Сохраните документ
5. Тезисно составьте конспект по данному занятию