

Урок 5. Списки, таблицы, iframe

Списки

Если на странице располагаются несколько элементов подряд друг под другом и имеет место перечисление, то используются списки.

Списки бывают трёх типов:

1. Маркированные (нумерованные)
2. Нумерованные
3. Списки определений

Последние используются очень редко, практически никогда.

Маркированный список

Для создания маркированного списка используется 2 тега: `` как контейнер и `` как элемент списка. Конструкция в целом выглядит таким образом:

```
<ul>
  <li>Первый элемент списка</li>
  <li>Второй элемент списка</li>
  <li>Третий элемент списка</li>
</ul>
```

Для создания вложенного списка необходимо в элемент `` поместить ещё один ``.

```
<ul>
  <li>Первый элемент списка</li>
  <li>Второй элемент списка</li>
  <li>Третий элемент списка
    <ul>
      <li>Первый элемент дочернего списка</li>
      <li>Второй элемент дочернего списка</li>
    </ul>
  </li>
</ul>
```

При этом у дочернего списка будет другой тип маркера и больший отступ слева для выделения.

В примере выше дочерний список находится внутри третьего элемента списка: так можно однозначно сказать, к какому конкретно элементу относится дочерний список.

Нумерованный список

В отличие от маркированного, в нумерованном списке каждый элемент имеет свой номер. Процесс создания списка отличается только главным тегом-контейнером: здесь используется ``:

```
<ol>
  <li>Первый элемент списка</li>
  <li>Второй элемент списка</li>
  <li>Третий элемент списка</li>
</ol>
```

Для создания дочернего списка используется аналогичный механизм:

```
<ol>
  <li>Первый элемент списка</li>
  <li>Второй элемент списка</li>
  <li>Третий элемент списка
    <ol>
      <li>Первый элемент дочернего списка</li>
      <li>Второй элемент дочернего списка</li>
    </ol>
  </li>
</ol>
```

При этом дочерний список не обязательно должен совпадать по типу с родительским: их можно совмещать. Внутри маркированного списка может быть нумерованный и наоборот.

У нумерованного списка есть два дополнительных атрибута: `start` и `reversed`

`reversed` используется без значения и если присутствует, то нумерация начинается с конца.

`start` используется для задания номера элемента, с которого необходимо начать отсчёт. Например, если `start` имеет значение `5`, то нумерация начнётся с 5 и продолжится далее: 6, 7, 8...

Пример использования этих атрибутов:

```
<ol start="5" reversed>
  <li>Первый элемент списка</li>
  <li>Второй элемент списка</li>
  <li>Третий элемент списка</li>
</ol>
```

В списке из примера выше нумерация начнётся с 5 (из-за значения атрибута `start` и пойдёт далее по убыванию: 4 и 3 (из-за присутствия атрибута `reversed`)).

Атрибут `type`

Как у нумерованного, так и у маркированного списка есть **устаревший** атрибут `type`. Он задаёт то, какие именно маркеры или цифры (буквы) будут отображаться слева от элемента списка.

Данный атрибут сейчас полностью заменён CSS свойствами.

Список определений

Список определений немного отличается по формированию от предыдущих двух и выглядит следующим образом:

```
<dl>
  <dt>Термин 1</dt>
  <dd>Описание термина 1</dd>
  <dt>Термин 2</dt>
  <dd>Описание термина 2</dd>
</dl>
```

Материалы

Дополнительно про списки можно прочитать здесь: <https://html5book.ru/html-lists/>

Таблицы

Для формирования простой таблицы используется несколько тегов:

1. `<table>` - как контейнер для всей таблицы
2. `<tr>` - как строка таблицы
3. `<th>` - как ячейка шапки таблицы
4. `<td>` - как обычная ячейка таблицы
5. `<caption>` - заголовок таблицы

```
<table>
  <caption>Студенты</caption>
  <tr>
    <th>Номер</th>
    <th>Имя<th>
    <th>Фамилия</th>
  <tr>
  <tr>
    <td>1</td>
    <td>Андрей</td>
    <td>Гулин</td>
  <tr>
  <tr>
    <td>2</td>
    <td>Иван</td>
    <td>Петров</td>
  <tr>
</table>
```

Так же в таблицах полезен атрибут `colspan` у ячеек: он позволяет объединить несколько столбцов в один.

```
<table>
  <tr>
    <td>Ячейка 1</td>
    <td>Ячейка 2</td>
    <td>Ячейка 3</td>
  </tr>
  <tr>
    <td>Ячейка 4</td>
    <td colspan="2">Ячейка 5</td>
  </tr>
</table>
```

В данном примере мы имеем таблицу с тремя столбцами. В первой строке будет три столбца и ячейки 1, 2 и 3. Во второй строке ячейка 4 будет расположена в первом столбце, а ячейка 5 будет занимать 2 столбца: второй и третий.

Материалы

У таблиц так же есть ещё некоторые теги, которые можно использовать внутри `<table>` и устаревшие атрибуты. Подробнее про них можно прочитать здесь: <https://html5book.ru/tablichnye-dannye/>

iframe

Тег `<iframe>` используется для вставки содержимого другой страницы на вашу страницу.

Чаще всего используется для встраивания видео или карт с ресурсов видеохостинга и сервисов карт соответственно.

Чаще всего сайты, позволяющие встраивать своё содержимое на страницы других сайтов, сами предоставляют целиком код `iframe`.

Для тестирования можно воспользоваться youtube: найти видео, открыть его, нажать кнопку “поделиться” и нажать кнопку “встроить”. Справа будет готовый код html для вставки на сайт.

Пример такого кода с youtube:

```
<iframe
  width="560"
  height="315"
  src="https://www.youtube.com/embed/JxS5E-kZc2s"
  title="YouTube video player"
  frameborder="0"
  allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-
picture"
  allowfullscreen>
</iframe>
```

Большинство современных сайтов не позволяет встраивать целиком страницы, но старые сайты позволяют и с этим можно поэкспериментировать. Например, можно

использовать сайт <http://aliweb.com> :

```
<iframe src="http://aliweb.com" width="600" height="400"></iframe>
```

Данная iframe возьмёт главную страницу сайта по адресу <http://aliweb.com> (с помощью атрибута `src` и нарисует рамку 600 пикселей в ширину и 400 пикселей в высоту с содержимым страницы сайта.

Рассмотрим другие атрибуты, использованные в iframe с youtube:

1. `title` - заголовок. Он не отображается, но используется для поисковых роботов или для голосовых помощников.
2. `frameborder` - указывает на размер в пикселях вокруг встраиваемого содержимого
3. `allow` - содержит список разрешений, которые указывают, какие действия на нашей странице может совершать встраиваемый контент.
4. `allowfullscreen` - при наличии разрешает открывать содержимое iframe во весь экран (если есть такая функция у содержимого)

Материалы

Подробнее об iframe можно почитать здесь: <https://webref.ru/html/iframe>