

Лекция 3. Основы работы с формами, встраивание аудио и видео

Добро пожаловать! Сегодня мы изучим основы работы с формами, а также встраивание аудио и видео. На первый взгляд может показаться, что формы — это просто поля для заполнения, а мультимедиа — лишь украшение страницы. Однако, давайте рассмотрим, почему эти элементы являются неотъемлемой частью веб-разработки.

Формы предоставляют механизм взаимодействия между пользователем и веб-страницей. Они не только собирают информацию, но и открывают двери для создания интерактивных и пользовательских интерфейсов. Благодаря формам, пользователи могут отправлять данные, оставлять комментарии и взаимодействовать с содержимым веб-сайта.

А что касается аудио и видео, то встраивание мультимедийных элементов придает страницам живость и динамичность. Это становится особенно важным в контексте современных требований к визуальному контенту. Аудиофайлы и видеоролики обогащают пользовательский опыт, делая веб-проекты более привлекательными и информативными.

Давайте вместе освоим основы работы с формами и научимся встраивать аудио и видео в ваши будущие веб-проекты.

Чтобы создавать интерактивные веб-страницы, нам понадобятся определенные теги. Основой для этого служит тег `<form>`. Этот тег выступает в роли контейнера, который объединяет различные элементы формы в единое целое.

Первым элементом, с которым мы познакомимся, будет тег `<input>`. Он представляет собой универсальное поле ввода, которое может быть использовано для разных целей, начиная от текстовых полей и заканчивая кнопками отправки.

Как будет выглядеть элемент `<input>` зависит от значения его атрибута `type` (тип). Например, поле ввода типа `"text"` используется для получения текстовой информации, а поле `"checkbox"` — для отметки нескольких вариантов. Разнообразие возможных значений атрибута `type` позволяет создавать разносторонние веб-формы, соответствующие различным потребностям пользователей:

- `text`: Создает текстовое поле для ввода.
- `password`: Поле для ввода пароля с маскировкой введенных символов.
- `checkbox`: Флажок для выбора (галочка).
- `radio`: Переключатель (радиокнопка).
- `submit`: Кнопка для отправки формы.
- `reset`: Кнопка для сброса значений формы.
- `file`: Выбор файла для загрузки.
- `number`: Поле для ввода числа.

- date: Поле для ввода даты.

Другие возможные атрибуты для тега input:

`name` (имя):

- Имя элемента формы, используется при отправке данных на сервер.

`value` (значение):

- Значение по умолчанию для поля ввода.

`placeholder` (подсказка):

- Текстовая подсказка, отображаемая внутри поля ввода до ввода данных.

`required` (обязательное):

- Сообщает браузеру, что поле должно быть заполнено перед отправкой формы.

`disabled` (отключено):

- Заблокировать элемент для взаимодействия с пользователем, он не будет реагировать на события.

`readonly` (только для чтения):

- Запретить редактирование значения элемента, оставляя его доступным для просмотра.

`autofocus` (автофокус):

- Автоматически устанавливает фокус в поле ввода при загрузке страницы.

Но все же не все возможные элементы формы создаются одним только тегом input. Тег ``<select>`` пригодится нам для создания выпадающих списков, а тег ``<textarea>`` станет инструментом для многострочного ввода текста. Обратим внимание на их особенности и то, как они взаимодействуют с пользователем.

Посмотрим на пример кода с тегом select:

```
<label for="fruits">Выберите ваш любимый фрукт:</label>

  <select id="fruits" name="fruits">
    <option value="apple">Яблоко</option>
    <option value="banana">Банан</option>
    <option value="orange">Апельсин</option>
    <option value="grape">Виноград</option>
    <option value="strawberry">Клубника</option>
  </select>
```

В нем стоит обратить внимание на то, что возможные опции для выбора в выпадающем списке оформляются специальными отдельными тегами `<option>`, а также на ранее еще не упомянутый тег `label` - это метка к полю, его название, показываемое в форме рядом с самим полем. Метка привязывается к полю с

использованием атрибутов for (в теге label) и id (в теге select). Метка никак браузером по-умолчанию не выделяется визуально - она выглядит как обычный текст в теге абзаца <p>, но ее наличие важно с точки зрения правильной структуры документа, его семантики (смысла), ну и также ее можно оформить и выделить как хочется с помощью стилей css, которые мы будем изучать далее.

Вот еще специальные возможные атрибуты для тега `<select>`:

`size` (размер):

- Определяет количество видимых строк в списке.

`multiple` (множественный выбор):

- Позволяет выбрать несколько вариантов.

Ну и также в нем можно использовать и общие для всех тегов формы атрибуты, такие как id, name, required, disabled, readonly и autofocus.

И последний из рассматриваемых нами сегодня тегов форм - `<textarea>` - он выглядит как многостроковое текстовое поле, в нем также возможно использовать те же общие атрибуты, что и для input и select. А возможные специальные атрибуты для него такие:

`rows` (количество строк):

- Определяет количество видимых строк в поле ввода текста.

`cols` (количество столбцов):

- Определяет количество видимых столбцов в поле ввода текста.

`placeholder` (подсказка):

- Текстовая подсказка, отображаемая внутри поля ввода до ввода данных.

Использование различных значений атрибутов тегов позволяет настраивать формы с учетом требований конкретного веб-проекта и предоставлять более удобный и функциональный пользовательский опыт. Практика с различными атрибутами поможет вам лучше овладеть их использованием в ваших будущих проектах.

Давайте перейдем к следующему этапу и углубимся в встраивание аудио и видео. Для этого мы используем тег `<audio>`, предоставляющий возможность легкого встраивания звуковых файлов в веб-страницы и создания мультимедийных веб-проектов.

```
<audio controls>
  <source src="audio/sample.mp3" type="audio/mp3">
  Ваш браузер не поддерживает встраивание аудиофайлов
</audio>
```

В приведенном примере тег ``<audio>`` содержит атрибут ``controls``, который добавляет на страницу элементы управления (панель воспроизведения). Также внутри тега ``<audio>`` мы включаем тег ``<source>``, который указывает путь к аудиофайлу и его тип.

Вы можете встроить аудиофайлы на свою веб-страницу, используя различные форматы, такие как MP3, WAV, или OGG. Это позволяет учесть различные браузеры и их поддержку аудиоформатов.

```
<audio controls>
  <source src="audio/music.mp3" type="audio/mp3">
  <source src="audio/music.ogg" type="audio/ogg">
  Ваш браузер не поддерживает встраивание аудиофайлов
</audio>
```

Важно учитывать, что поддерживаемые форматы могут различаться в разных браузерах. В примерах мы использовали MP3, OGG, но также можно добавить WAV. При этом параметры, такие как ``autoplay`` (автоматическое воспроизведение), ``loop`` (петля), и другие, могут дополнительно настраивать поведение аудио.

Синтаксис и использование тега ``<video>`` очень похоже на тег ``<audio>`` - кроме аудио вы можете также легко встроить видеофайлы.

```
<video controls width="640" height="360">
  <source src="video/sample.mp4" type="video/mp4">
  Ваш браузер не поддерживает встраивание видео
</video>
```

В этом примере тег ``<video>`` содержит атрибут ``controls`` для добавления элементов управления, а также атрибуты ``width`` и ``height`` для определения размеров видео. Тег ``<source>`` указывает путь к видеофайлу и его формат.

Давайте посмотрим на другой пример встраивания видео с различными параметрами:

```
<video controls width="800" height="450" autoplay loop>
  <source src="video/movie.mp4" type="video/mp4">
  <source src="video/movie.webm" type="video/webm">
  Ваш браузер не поддерживает встраивание видео
</video>
```

Здесь видео будет воспроизводиться автоматически при загрузке страницы (атрибут ``autoplay``) и зацикливаться (атрибут ``loop``), создавая эффект бесконечного воспроизведения.

Приступайте к встраиванию аудио и на свои страницы и экспериментируйте с различными параметрами для достижения желаемого результата.

Поговорим о том, где вы можете найти узнать какие возможны параметры и какие в них можно поставить значения и другие необходимые сведения о HTML. При

разработке веб-страницы невероятно важно уметь использовать официальную (да и не официальную) документацию для получения подробной информации о тегах, атрибутах и других элементах языка разметки.

Где искать документацию:

Официальная документация HTML всегда доступна онлайн на сайте W3C (Консорциума Всемирной паутины). Просто перейдите на сайт и найдите раздел, посвященный HTML - <http://www.w3.org/TR/html5/>.

Если официальная документация вам кажется сложно и непонятно написанной, то я рекомендую сайт <https://htmlbook.ru/html> - там есть справочник всех тегов и атрибутов, поиск, самоучители, статьи и другая полезная информация.

Тег <video>

19.10.2010						Влад Мерзевич
Internet Explorer	Chrome	Opera	Safari	Firefox	Android	iOS
✓ 9.0+	✓ 3.0+	✓ 10.50+	✓ 3.1+	✓ 3.5+	✓ 2.0+	✓ 2.0+

Спецификация

HTML:	3.2	4.01	5.0	XHTML:	1.0	1.1
-------	-----	------	-----	--------	-----	-----

Описание

Добавляет, воспроизводит и управляет настройками видеоролика на веб-странице. Путь к файлу задается через атрибут **src** или вложенный тег [<source>](#). Список поддерживаемых браузерами аудио и видеокодеков ограничен и приведен в табл. 1.

Когда вам нужно подробное описание атрибутов и их значений, документация предоставляет исчерпывающую информацию. Вы можете узнать, как использовать каждый атрибут и как он взаимодействует с другими элементами. Также из документации можно узнать в какой версии стандарта html поддерживается тот или иной тег и атрибут, особенности его поддержки различными версиями браузеров.

Табл. 1. Кодеки и браузеры

Браузер	Internet Explorer	Chrome	Opera	Safari	Firefox
Аудио кодеки					
ogg/vorbis	✗	✓	✓	✗	✓
wav	✗	✗	✓	✓	✓
mp3	✓	✓	✗	✓	✗
AAC	✓	✓	✗	✓	✗
Видео кодеки					
ogg/theora	✗	✓	✓	✗	✓
H.264	✓	✓	✗	✓	✗
WebM	✗	✓	✓	✗	✓

Не забывайте, что помимо изученных тегов, существует множество других, которые могут пригодиться в различных сценариях. Например, тег ``<table>`` для создания таблицы, другие теги для работы с формами, а также множество других, которые увидите при более глубоком изучении документации.

Справочник по HTML

23.09.2010

Влад Мерзевич

I

- [<!DOCTYPE>](#)
- [<!-- -->](#)

A

- [<a>](#)
- [<abbr>](#)
- [<acronym>](#)
- [<address>](#)
- [<applet>](#)
- [<area>](#)
- [<article>](#) HTML5
- [<aside>](#) HTML5
- [<audio>](#) HTML5

B

- [](#)
- [<base>](#)
- [<basefont>](#)
- [<bdi>](#) HTML5
- [<bdo>](#)
- [<bgsound>](#)
- [<blockquote>](#)
- [<big>](#)
- [<body>](#)
- [<blink>](#)
- [
](#)
- [<button>](#)

C

- [<canvas>](#) HTML5
- [<caption>](#)
- [<center>](#)
- [<cite>](#)
- [<code>](#)
- [<col>](#)
- [<colgroup>](#)
- [<command>](#) HTML5
- [<comment>](#)

Типы тегов

- HTML5
- Блочные элементы
- Строчные элементы
- Универсальные элементы
- Нестандартные теги
- Осуждаемые теги
- Видео
- Документ
- Звук
- Изображения
- Объекты
- Скрипты
- Списки
- Ссылки
- Таблицы
- Текст
- Форматирование
- Формы
- Фреймы

Приобретение навыков поиска документации в Интернете и работы с ней — ключ к эффективной веб-разработке. Не стесняйтесь обращаться в Google, Яндекс и к рекомендуемой мной документации в процессе работы — она всегда вам подскажет что-то новое, что вы сможете использовать в своих веб-проектах.

Итак, в этой лекции мы освоили основы работы с формами и встраивания мультимедиа в HTML, а также узнали как с помощью документации и Интернета вы можете узнать больше. Практикуйтесь, экспериментируйте и не забывайте заглядывать в документацию для открытия новых возможностей. Желаю успехов вам в вашем развитии как веб-разработчика!