

WebWorker. Audio. Video

№ урока: 3 **Курс:** HTML5&CSS3 Advanced

Средства обучения: Visual Studio, Visual Studio Code, NotePad++

Обзор, цель и назначение урока

Цель урока – ознакомление с элементом **WebWorkers** для подключения дополнительных потоков данных на страницу. Научиться использовать дополнительные функции и свойства аудио и видео элементов HTML5.

Изучив материал данного занятия, учащийся сможет:

- Применять на странице **WebWorker**.
- Применять различные виды медиаконтейнеров в верстке страницы.
- Добавлять на страницу аудио и видеофайлы, указав несколько источников.
- Использовать на странице собственные элементы управления проигрывателем.

Содержание урока

1. Использование многопоточности на странице.
2. Использование **XMLHttpRequest** в связке с **WebWorker**.
3. Встраиваемое аудио.
4. Свойства и методы работы с аудио.
5. Встраиваемое видео.
6. Способы отображения интерфейса и проигрывания, встроенного видео. Атрибут **Controls**.

Резюме

- **WebWorkers** – технология, позволяющая устанавливать специальную среду кода, где можно выполнять задания в отдельном потоке данных, не зависящем от основного кода. В отдельные файлы выносятся функции, обеспечивающие функционирование **AJAX**, либо функции обрабатывающие большие массивы информации, которые во время работы уменьшают скорость построения страницы. При выполнении скрипта на стороне браузера создается специальный объект **Worker**, который и отвечает за вызов необходимых функций.
- Существующий **Worker** может отсылать сообщения в **JavaScript** код, который его создал, отправляя сообщения в обработчик событий, указанный этим кодом (и наоборот).
- **Worker** - это объект, созданный при помощи конструктора, который запускает **JavaScript** файл — содержащий код, который будет выполнен в потоке **Worker**; объекты **Workers** запускаются в другом глобальном контексте, который отличается от текущего **window**. Таким образом, использование переменной **window** для получения текущего глобального контекста (вместо **self**) внутри **Worker** вернет ошибку.
- **Worker** элементов может быть сколько угодно и все они могут работать одновременно, при этом формирование страницы не будет остановлено, в ожидании пока отработает, какая-либо функция.
- Чтобы завершить работу **worker** и освободить ресурсы компьютера Вы можете использовать метод **terminate()**.
- **<audio>** – встраивание аудио на веб-страницу. Атрибуты:
- **Autoplay** - звук начинает играть сразу после загрузки страницы.
- **Controls** - Добавляет панель управления к аудиофайлу.
- **Loop** - Повторяет воспроизведение звука с начала после его завершения.
- **Preload** - Используется для загрузки файла вместе с загрузкой веб-страницы.
- **Src** - Указывает путь к воспроизводимому файлу.

- **Muted** -Логический атрибут, который указывает, будет ли изначально заглушено воспроизведение. По умолчанию **false**.
- **Volume** - Громкость воспроизведения, в диапазоне от 0.0 (самая тихая) до 1.0 (самая громкая).
- **Play()** – метод **js** для запуска аудио дорожки.
- **Pause()** - метод **js** для остановки аудио дорожки.
- **CurrentTime** – текущее время аудио.
- **Duration** – длительность аудио в секундах.
- **<video>** – встраивание видео на веб-страницу.
- В мультимедийном контейнере определен только способ хранения содержимого, а не тип этого содержимого.
- Популярные форматы видеоконтейнеров: **MPEG-4, Flash Video, WebM, Ogg**.
- Видеокодек – это алгоритм кодировки видеопотока. Популярные форматы: **H.264, Theora** и **VP8**.
- Один тег **<video>** может ссылаться на несколько видеофайлов; из них браузер выберет самый первый, который сумеет воспроизвести.
- **Controls** – способ отображения интерфейса проигрывателя.
- Атрибут **autoplay** говорит браузеру, что надо не только начать скачивать видео сразу по окончании загрузки страницы, но и начать его воспроизводить как можно раньше.
- Атрибут **preload** говорит браузеру, что скачивание видео надо начать сразу по окончании загрузки страницы.
- **Poster** - **URL**-адрес, указывающий на постер, который будет использован, пока загружается видео или пока пользователь не нажмет на кнопку воспроизведения. Если этот атрибут не указан, ничего не отображается до тех пор, пока не будет доступен первый кадр.
- **PlaybackRate** - свойство, указывающее уровень скорости воспроизведения.

Закрепление материала

- Как создать новый **Worker** на странице?
- На какое свойство реагирует при передаче данных **Worker**?
- Какой метод отправляет данные между потоками?
- Какое свойство позволяет увеличить громкость аудио?
- Какое свойство позволяет изменить скорость воспроизведения видео?
- Как узнать длительность аудиофайла?

Дополнительное задание

Добавьте на страницу окно для просмотра видео и кнопки:

- остановить,
- продолжить просмотр,
- вперед на 10с,
- назад на 10 с,
- увеличить громкость,
- уменьшить громкость.

Также добавьте на страницу 3 кнопки с ссылками на видеофайлы. При нажатии на выбранный ролик, в окне начинается просмотр выбранного пользователем. После перезагрузки страницы, видео должно загрузиться, и пользователь может продолжить просмотр (который был до перезагрузки). Используйте сохранение информации в хранилище.

Самостоятельная деятельность учащегося

Задание 1

Выучить основные понятия, рассмотренные на уроке.

Задание 2

Создайте сайт, где будет возможность производить сложный расчет данных из калькулятора.

Добавьте 2 поля ввода и 4 кнопки для математических операций. После нажатия на каждую из кнопок – должен создаваться свой Worker и при обращении к потоку – в нем будет производиться вычисления. Используйте 4 файла Worker.js с различным функционалом.

Рекомендуемые ресурсы

<http://htmlbook.ru/>

<https://html.com/>

<https://www.html5rocks.com/>