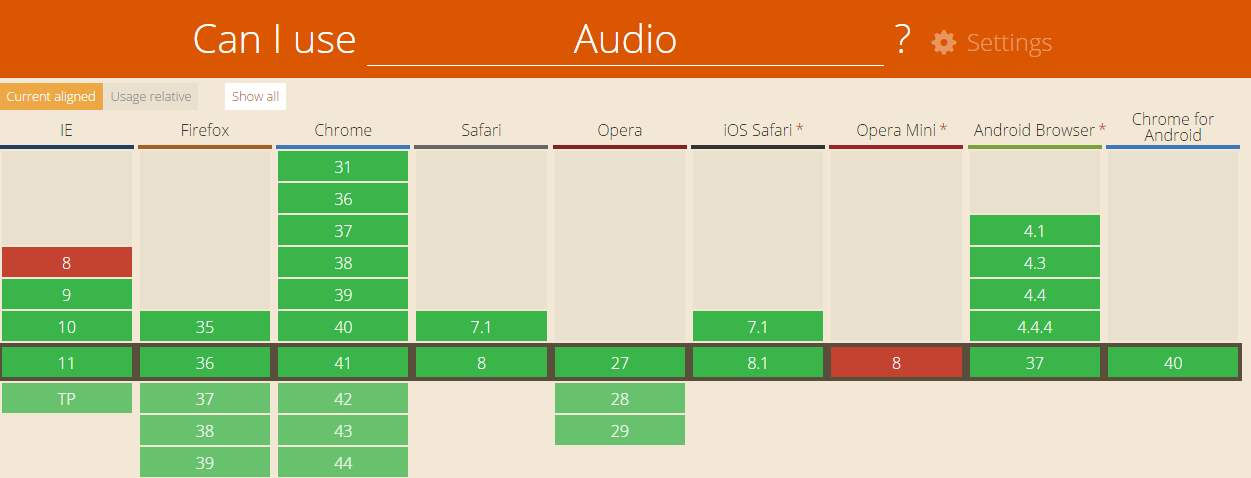
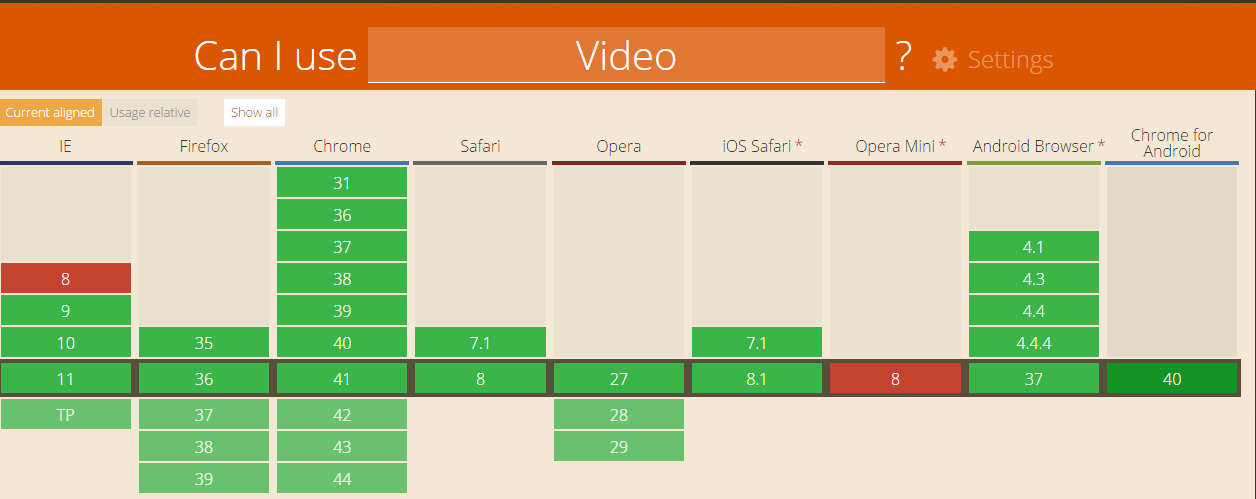
Audio & Video HTML5 API





Добавление аудио элемента:

<audio src=" URL или Путь к файлу">

<p>Your browser does not support the audio element.</p>

</audio>

Добавление видео элемента:

<video src="URL или Путь к файлу" controls>

the <code>video</code> element.

</video>

*Специальные атрибуты:*

- controls: добавляет панель управления (паузу, звук..);

- autoplay: проигрывает автоматически;

- loop: проигрывает циклически;

- preload: используется для больших файлов, инициирует «раннюю» загрузку, имеет 3 значения: “none” – файл не загружается до востребования;

“auto” – файл начинает загружаться еще до востребования;

“metadata” – заранее загружаются только метаданные файла.

Для поддержки медиа в разных браузерах можно добавлять видео разных форматов:

<video controls>

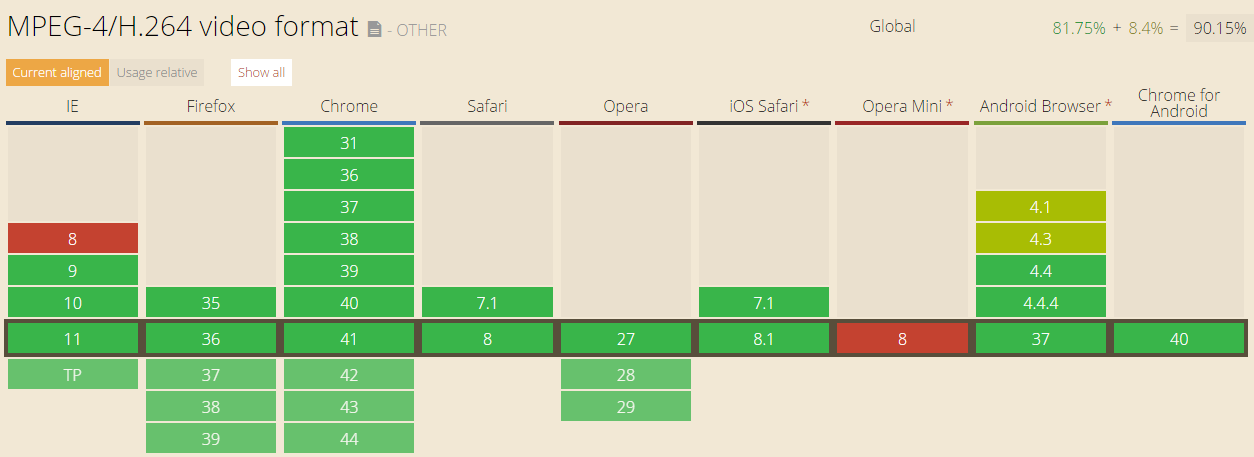
<source src="foo.ogg" type="video/ogg">

<source src="foo.mp4" type="video/mp4">

Your browser does not support the <code>video</code> element.

</video>

Из всех форматов лучше всего поддерживается MPEG-4



Медиа – элементы также можно контролировать с помощью JavaScript

var v = document.getElementsByTagName("video")[0];

v.play();

<audio id="demo" src="audio.mp3"></audio>

<div>

<button onclick="document.getElementById('demo').play()">Play the Audio</button>

<button onclick="document.getElementById('demo').pause()">Pause the Audio</button>

<button onclick="document.getElementById('demo').volume+=0.1">Increase Volume</button>

<button onclick="document.getElementById('demo').volume-=0.1">Decrease Volume</button>

</div>

Обращаясь к медиа-элементам через скрипт можно получить дополнительные сведения:

var mediaElement = document.getElementById('mediaElementID');

mediaElement.seekable.start(); // Returns the starting time (in seconds)

mediaElement.seekable.end(); // Returns the ending time (in seconds)

mediaElement.currentTime = 122; // Seek to 122 seconds

mediaElement.played.end(); // Returns the number of seconds the browser has played

Добавляя элемент на страницу, можно указывать какой временной промежуток должен быть проигран:

#t=[starttime][,endtime]

**url\_или\_путь\_к\_файлу#t=10,20** - воспроизведется отрезок с 10 по 20 секунду

**url\_или\_путь\_к\_файлу #t=,10.5** – воспроизведется от начала и до 10.5 seconds.

**url\_или\_путь\_к\_файлу #t=,02:00:00** – от начала и до 2 часов, если полная продолжительность меньше 2 часов, то воспроизведет 1 раз до конца и все.

**url\_или\_путь\_к\_файлу #t=60** – начнет воспроизводиться с 60 секунды и до конца

Чтоб определить загрузилось ли видео, можно проверять свойство networkState

0 = NETWORK\_EMPTY - audio/video has not yet been initialized

1 = NETWORK\_IDLE - audio/video is active and has selected a resource, but is not using the network

2 = NETWORK\_LOADING - browser is downloading data

3 = NETWORK\_NO\_SOURCE - no audio/video source found

В случае, когда браузер не поддерживает html5, можно предоставить альтернативу:

- флеш-видео

<video src="video.ogv" controls>

<object data="flvplayer.swf" type="application/x-shockwave-flash">

<param value="flvplayer.swf" name="movie"/>

</object>

</video>

Сделать респонсив видео:

Добавить респонсив видео можно с помощью iframe:

<iframe src="http://www.youtube.com/embed/4aQwT3n2c1Q" height="315" width="560" allowfullscreen="" frameborder="0"></iframe>

Для того чтоб обойти тот факт, что на iframe нужно задавать ширину и высоту, его оборачивают в дополнительный <div class=”video-container”> </div>.

.video-container {

position: relative; // чтоб позиционировать iframe относительно этого блока

padding-bottom: 56.25%; // 16/9 –отношение ширины к высоте

padding-top: 35px;// - необходимо для youtube видео

height: 0;// высота задается паддингами

overflow: hidden;// чтоб скрыть то, что может сломать лейаут

}

Стили для iframe

.video-container iframe {

position: absolute;

top:0;

left: 0;

width: 100%;

height: 100%;

}

Управление видео элементами с помощью javascript:

Проверка поддерживает ли браузер video

var supportsVideo = !!document.createElement('video').canPlayType;

Доступен ряд свойств, с помощью которых можно отслеживать и контролировать текущее состояние элемента

Скрыть встроенную панель управления:

video.controls = false;

Проверка, остановлено ли видео

video.paused

Закончилось ли

video.ended

Начать воспроизведение видео

video.play();

Остановить видео

video.pause();

Манипулировать с какого времени воспроизводить

video.currentTime = 0;

Воспроизводить звук или нет

video.muted = !video.muted;

Текущее значение громкости: (от 0 до 1)

video.volume

Продолжительность видео

video.duration

Элементы [<track>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/track) прикрепленные к видео

video.textTracks[i]

Доступные события:

Loadedmetadata – метаданные видео загружены

Timeupdate – инициируется периодически во время проигрывания видео

Captions versus subtitles

Реализуются с помощью тега  [<track>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/track).

Файл, содержащий нужный текст, должен быть формата WebVTT ( [Web Video Text Tracks](https://developer.mozilla.org/en-US/docs/HTML/WebVTT) )

Каждый [<track>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/track) имеет набор атрибутов:

- kind – тип контента (например captions, **subtitles , descriptions, chapters, metadata**)

- label – указывает язык контента (label="English")

- src – url к vtt файлу

- srclang – (srclang="en")

- default – [<track>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/track) элемент, который сработает по умолчанию