






Mit HTML 5 kann man auch Audio- und Videodateien einbinden. Ratsam ist es, mit Multimediaelementen gut überlegt umzugehen. Automatisch abgespielte Soundfiles können den Benutzer entnerven – große Dateien brauchen selbstverständlich mehr Ladezeit.

Browsersupport					
<audio>	4.0	9.0	3.5	4.0	10.5

HTML <audio> </audio>



Es können Musik und Soundfiles in den Formaten .mp3, .wav und .ogg abgespielt werden. Wobei .mp3 den Vorzug haben sollte, weil dieses Format von den meisten Browsern unterstützt wird.

Folgende Attribute sind für das <audio> Element verfügbar:

autoplay	beginnt sofort mit dem Abspielen	ohne Wert
controls	zeigt die Steuerungsfunktion an (Lautstärke usw.)	ohne Wert
loop	das Abspielen wird endlos wiederholt	ohne Wert
muted	zu Beginn stummschalten	ohne Wert
preload	definiert das Ladeverhalten:	
	auto ← die gesamte Datei wird geladen	
	none ← die Datei wird nicht vorgeladen	
	metadata ← es werden nur die Metadaten geladen	
src	URL zur Audiodatei	

Zusätzlich gibt es noch TimeRange Attribute die für JavaScript von Bedeutung sind:

buffered	liest aus, wieviel zwischengespeichert wurde
played	liest aus, wie viel schon gespielt wurde.

Der Text zwischen dem <audio> Tag wird angezeigt, wenn das Abspielen nicht möglich ist. z. B. weil es in den Browsereinstellungen deaktiviert wurde.

```
<audio controls loop src="Europahymne.mp3">
  Keine Audiowiedergabe möglich!</audio>
```



Ein Fallback ist eine Alternative die Abgespielt werden soll, wenn das primäre Sound-Format nicht abgespielt werden kann.

Dafür wird im <audio> Element weitere <source> Tags setzen:

```
<audio controls autoplay="true" >
  <source src="Europahymne.mp3" type="audio/mp3" />
  <source src="Europahymne.ogg" type="audio/ogg" />
  Keine Audiowiedergabe möglich!
</audio>
```