

HTML5 Audio

ATTENTION

Quotes in the provided codes in this tutorial must be **regular quotes**.

Regular (straight) quotes are frequently converted to smart (curly) quotes in word processors and PDFs .

Regular quote: " " . . .

Smart quote: “ ” . . .

To avoid this, directly type code or copy-paste it in a plain text editor (such as Notepad, Gedit...) and then copy-paste it from the plain text editor into Hot Potatoes.

There is **two ways** to insert HTML5 audio player in Jmatch add-ons:



HTML5 mini audio button

Classic HTML5 audio player control

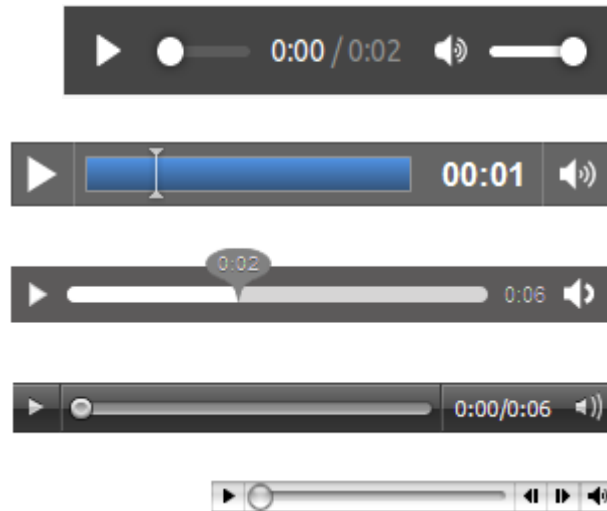
I Classic HTML5 audio player

1- Insertion

Use the possibility given by Hot Potatoes v7: Insert> Insert a media object.

2- Appearance

Every browser has its own design for the HTML5 audio player control:



As you can see, the size and colours of the player control varies from one to another.

There is no way to change the size or the colours.

[More information.](#)

II HTML5 mini audio button

Most of the time we use very short audio files (one word or a short sentence) so the classic audio player control is too large. A minimalist HTML5 audio player was created for my add-ons.

Its control bar offers only a play button (or possibly a play/pause button).



The player will accept **mp3**, **ogg** and **wav** extensions for audio files.

You can use it in every part of the exercise : title, subtitle, items, reading text.

1- How to insert an audio file with a mini button in Jmatch add-ons?

You need **player3.js** in your exercise folder. You will find it in the demo folder of the add-ons.

a) New method

- Insert an audio file via **Insert> Insert a media object**.

You will get something like that:

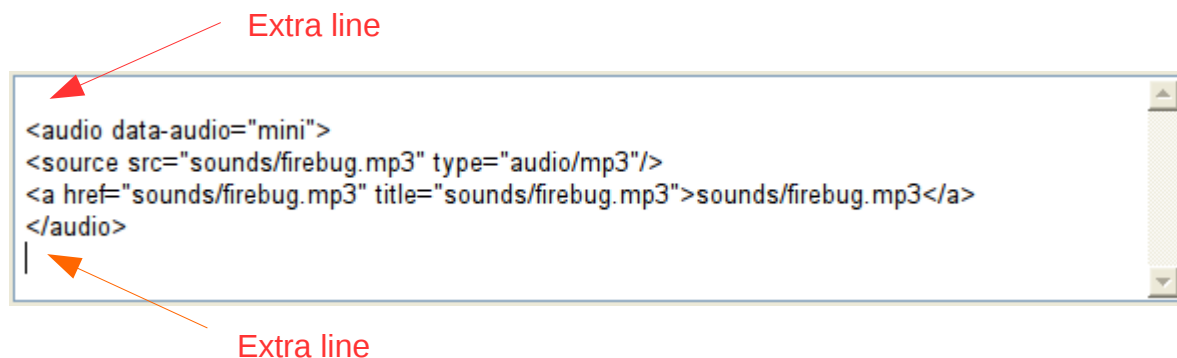
```
<audio controls="controls">
<source src="myaudiofile .mp3" type="audio/mp3"/>
<a href="myaudiofile.mp3" title="myaudiofile.mp3">myaudiofile</a>
</audio>
```

where myaudiofile.mp3 is an audio file in the exercise folder.

- To have a mini audio button instead of the classic HTML5 audio player control just **remove controls="controls"** and add **data-audio="mini"** inside the audio tag:

```
<audio data-audio="mini">
<source src="myaudiofile .mp3" type="audio/mp3"/>
<a href="myaudiofile.mp3" title="myaudiofile.mp3">myaudiofile</a>
</audio>
```

- **Remove the lines** inserted by Hot Potatoes **before and after the code**:



If you don't remove the lines you will get  instead of .

Note: It's difficult to know if there is an extra line after the code. If you can put your cursor on the line under the code (as on the image above), there is one that you have to remove.

That's all!

b) Compatibility with the old method

- The old method (for Hot Potatoes v6) can still be used:

```
<a href="javascript:void(0);" onclick="play('myaudiofile.mp3');"
class="gloss">&#9658;</a>
```

- It means that you don't have to modify your old jmt exercises where this method was used.

Be aware that, on touch devices, this mini audio button is sometimes distorted (not a circle).

2- How to have a pause button?

Usually we use only words or short sentences as sounds. For these, a pause button is not useful. If your sound file is bigger, you would like a play/pause button.



Open **player3.js** with a text editor such as notepad or notepad++, gedit etc. (don't use Microsoft Word or Libreoffice).

At the beginning of the file you will see:

```
var unicode_play='&#9658;';

//var unicode_pause='<span style="display:inline-block;font-family="Arial";font-size:65%;vertical-align:middle;padding:0px 2px 3px 2px;">|&nbsp;&nbsp;&nbsp;|</span>';
// for pause button : uncomment the line above and comment the line below
var unicode_pause='&#9658;';
```

Change it as mentioned:

To comment: add "//" at the beginning of the line.

To uncomment: delete "//" at the beginning of the line.

After the change, it should look like this:

```
var unicode_play='&#9658;';

//var unicode_pause='<span style="display:inline-block;font-family="Arial";font-size:65%;vertical-align:middle;padding:0px 2px 3px 2px;">|&nbsp;&nbsp;&nbsp;|</span>';
// for no pause button : comment the line above and uncomment the line below
//var unicode_pause='&#9658;';
```

Be aware that, on touch devices, this pause button is sometimes distorted.

Don't forget to **save the modified file**.

3- How to change the style of the audio button?

The style is defined in **hp7.cs_**.

Colors

- **Open hp7.cs_** with a text editor such as notepad or notepad++, gedit etc.(don't use Microsoft Word or Libreoffice) and search for "**--strGloss**". You will find it near line 70.

```
--strGlossColor: green; /* for mini audio button */  
--strGlossBGColor: gold; /* for mini audio button */
```



- You can enter any valid css values in place of the original values.
For example:

```
--strGlossColor: blue; /* for mini audio button */  
--strGlossBGColor: silver; /* for mini audio button */
```



You can find the names or the HTML hexadecimal codes for colours on [wikipedia](https://www.wikipedia.org).

Don't forget to save the modified file.

Shape

- The values of padding in the gloss class are perfect for a circular button on a PC (tested with Firefox and Chrome).

```
padding: 0.5em 0.5em 0.5em 0.5em; /*ok for pc, font-family : Arial*/
```

- Unfortunately they are not adapted to Android (and ios?) devices.



On a Samsung with Android v9 these values are correct:

```
padding: 0.5em 0.9em 0.5em 0.9em; /*ok for android*/
```



You must make your own tests. To modify the padding, **open hp7.cs_** with a text editor such as notepad or notepad++, gedit etc.(don't use Microsoft Word or Libreoffice) and search for "**a:active.gloss**". You will find it near line 795.

Don't forget to save the modified file.

Other changes

Open hp7.cs_ with a text editor such as notepad or notepad++, gedit etc.(don't use Microsoft Word or Libreoffice) and search for "**Audio Gloss style**". You will find it near line 810.

Don't forget to save the modified file.