

# メディアプレイヤー

今回は、音声と映像の再生について学習します。

音声映像の再生には、これまでさまざまなブラウザが独自の機能を搭載させてきました。  
また、複数のブラウザで共通に再生するためには「プラグイン」による再生が一般的でした。メディア再生以外にも独自に発展するブラウザを W3C は HTML5 を発表し、統一しました。

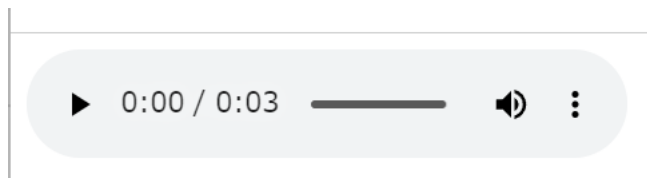
これにより、制作者はブラウザごとに異なる再生方法のコード記述に気を使うことなく、音声や動画を埋め込むことができるようになりました。W3C が HTML5 で推進した「標準化」の良い現われです。

## ■音声の再生

HTML5 では、音声作成のタグ<audio>要素が用意されました。  
使い方は非常に簡単です。

```
<audio src="./sample.mp3" controls>
</audio>
```

これは HTML タグなので、このタグを<body>の中に記述します。



## ◆audio 属性

**control** : 再生・停止等のコントローラーが表示されます

※この指定がないとブラウザ上には何も表示されません。

**autoplay** : 自動再生

**preload** : auto で自動読み込み

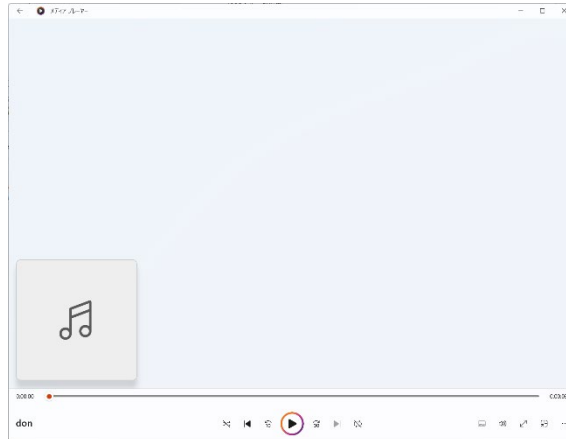
**loop** : ループ再生

## ■JavaScript での audio 指定

JavaScript から audio タグを使うためには、事前に「ファイル名」と「id」を指定する必要があります

```
var audio=new Audio("ファイル名");  
var audio=document.getElementById("id 名");
```

※id 名は、何でもかまいません。



## ◆使用可能な音声コーデック

MPEG-3、AAC、Ogg Vorbis（Google 社開発）、WAV（Microsoft & IBM 社開発）

## ◆JavaScript で audio 再生する利点

audio の再生だけであれば、はじめに学習した HTML でも可能です。JavaScript で audio 再生する利点は、計算や条件分岐の結果で「音を出す」などの処理が必要な場合です。

例えば、勝ったら「ウァ〜オ！！」、負けたら「チ〜ン！」とかを音を出すことができます。

```
var music = new Audio('abc.mp3');  
music.play();  
music.pause();
```

## ◆自動再生されない

しかし実は、JavaScript で音声再生は少し大変です。  
そもそも、現在のブラウザでは「自動再生」してくれません。

無条件に再生してくれるのは、

- ・ホワイトリストの掲載など安全性を事前に認めている場合
- ・クリックなどのイベントによる場合
- ・再生音量が 0 またはミュートになっている場合

などの場合。

そこで、ここではもっとも一般的に使われるイベント（「クリック」）による再生起動を見ていきます。

## ◆プリロード

音楽ファイルや映像ファイルは、ファイルサイズが大きいため、プログラムが起動してから読み込んでいてはすぐに動かない場合があります。そこで、事前にファイルを読み込んでおく処理（プリロード）をします。

```
<audio id="snd1" preload loop>  
<source src="./chin.mp3" type="audio/mp3">  
</audio>
```

読み込む音声ファイルに id を付けます。（ここでは“snd1”）

次に、音声ファイルを指定します。「./」は、このソースがあるディレクトリと同じディレクトリにソースがあることを明示しています。

さらに、オーディオのタイプを指定しています。上で挙げた「使用可能な音声コーデック」を指定します。OS 同士で、互いに受け入れないファイル形式があるので、その場合には、Windows 用と MAC 用と同じ内容のファイルを 2 つ用意して、ここで指定する必要があります。（面倒なんですけど〇〇さん！）

## ■動画再生

動画は<video>タグで音声と同様に手軽にあつかえます

```
<video src="sample.mp4" controls preload="auto" width="400" height="300"
poster="img/sample.jpg"></video>
```

## ◆video 属性

width、height：動画再生では、再生する画面サイズを指定します

poster：動画再生前のサムネイルの指定

そのほか、<audio>と同じように使えます

## ◆各動画形式の対応

さまざまな動画形式に対応する場合は、<source>指定で並記します

```
<video controls>
  <source src="abc.mp4">
  <source src="abc.ogv">
</video>
```

この表記だと、Firefox では「ogv」形式、Safari だと「mp4」形式が読み込まれます。

## ◆使用可能な動画フォーマット

avi (Microsoft 開発)、flv (Adobe Flash 配信用)、MPEG-4、mkv、ogg

## ◆動画のコーデック

H.264、Ogg、VP8

Apple 社は QuickTime などへのリソースを H.264 に大量に投入しているため、H.264 を推奨しています。また、Firefox などオープンリソースを目指すブラウザは、同じくオープンな Ogg を推奨しています。このように各ブラウザに対応させるためには、<source>と複数行書く必要があり、また、動画は複数形式準備する必要があります。