

## 13강. 오디오와 비디오

### 1. 오디오

#### 1.1 기존방법

```

<html>
<head>
<title> 오디오/비디오 삽입 </title>
</head>
<body>
<embed src="audio.wav">
</body>
</html>

```

플러그인을 사용하여 표시되는 데이터나 동영상을 웹 페이지에 삽입

```

<audio src="~"> ... </audio>
<video src="~"> ... </video>

<object data="URL" ~ </object>

```

- embed 요소를 사용하여 다음과 같이 사용한다.

```
<embed src="musicvideo.avi" width="300" height="300">
```

#### 1.2 오디오 요소: audio

- 웹 브라우저에서 플러그인의 도움 없이 오디오 재생할 수 있다.

```

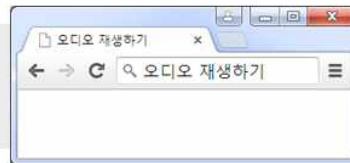
<audio src="오디오 파일 주소">
  audio 요소를 지원하지 않는 브라우저에서 대체 표시할 내용
</audio>

```

```

<audio src="song.mp3">
  <embed src="song.mp3"></embed>
</audio>

```



- 오디오를 재생하기 위해서는 src 속성을 사용하여 재생할 오디오 파일의 경로 지정한다. 재생 컨트롤의 집합(재생, 일시 정지, 볼륨 등)의 표시 여부를 지정하도록 controls 속성을 사용하고 웹 문서 로딩 시에 오디오를 자동으로 재생할 지 여부를 autoplay 속성을 사용한다. 오디오가 반복적으로 재생되도록 loop 속성을 사용한다.
- preload는 웹 문서 로딩 시에 오디오의 로딩 상태를 나타내며 none, metadata, auto 값을 사용할 수 있다.
- controls, autoplay, loop 속성은 별도로 값을 지정하지 않아도 무방하다.

### 1.3 브라우저별 지원하는 오디오 포맷

오디오 포맷	브라우저				
	크롬 37	IE 11	FF 32	오페라 24	사파리 8
PCM	○	X	○	○	X
AAC	○	○	X	X	○
MP3	○	○	○	X	○
Ogg Vorbis	○	X	○	○	X
Ogg Opus	○	X	○	○	X
WebM (Vorbis)	○	X	○	○	X
WebM (Opus)	○	X	○	○	X

### 1.4 source 요소

- audio/video 요소의 src 속성의 역할을 대신하는 요소로 속성들은 src, type, media가 있다.
- 미디어 요소의 자식 요소로서, 여러 개 지정 가능하며 차례대로 source 요소를 실행하다가 지원하는 포맷을 찾아서 재생 한다.

```

<audio controls>
  <source src="오디오 파일 주소" type="MIME 타입">
  <source src="오디오 파일 주소" type="MIME 타입">
  ... .. (or 미지원 브라우저에서의 대체 표현)
</audio>

```

↓

```

<audio controls>
  <source src="audio_sample.ogg" type="audio/ogg">
  <source src="audio_sample.mp3" type="audio/mpeg">
  현재의 브라우저는 audio 요소를 지원하지 않습니다.
</audio>

```

### 1.5 브라우저에서 재생 가능한 오디오 포맷 검사 방법

- 오디오 요소를 가져온다.

```

var audio = document.getElementById('오디오 아이디');
var audio = new Audio( );

```

- 오디오 포맷을 지원하는지 canPlayType() 메서드를 사용하여 조사한다. canPlayType() 메서드를 사용하면 probably, maybe, "" (빈 문자열) 중에 하나를 반환한다.

```
audio.canPlayType('audio/ogg');
audio.canPlayType('audio/ogg;codecs="vorbis" ');
```

- 오디오 컨테이너, 포맷, 파일 확장자, MIME 타입, 코덱 문자열

컨테이너	포맷	파일 확장자	MIME 타입	코덱 문자열
MP3	MP3	.mp3	audio/mpeg	mp3
MP4	AAC, AAC+	.mp4, .m4a, .aac	audio/mp4	mp4a.40.5
OGA/OGG	Ogg Vorbis	.oga, .ogg	audio/ogg	vorbis
WAV	PCM	.wav	audio/wav	1

### 1.6 미디어(오디오, 비디오) 제어

- 자바스크립트를 통한 미디어 파일(오디오, 비디오)의 제어한다.
- 미디어 파일 제어 관련 메서드는 load(), play(), pause()가 있다.

#### 1.6.1 미디어 파일 제어 및 상태 관련 속성

src	미디어 파일의 경로를 지정한다.
currentSrc	재생 중인 미디어 파일의 URL을 반환한다.
currentTime	현재 재생 위치를 초 단위로 반환한다.
duration	미디어 파일의 총 재생 시간을 초 단위로 반환한다.
paused	일시 정지 상태인지의 여부를 반환한다.
defaultPlaybackRate	기본 재생 속도를 반환/지정 한다.
playbackRate	현재 재생 속도를 반환/지정 한다.
ended	재생이 종료되어 있는 상태인지 여부 반환한다.
autoplay	미디어 파일이 자동 재생되도록 지정한다.
loop	반복 재생되도록 지정한다.
controls	플레이어의 출력 여부 지정한다.
volume	볼륨을 지정하거나 상태 반환한다.
muted	음소거 상태 반환한다.
error	오류 정보 반환한다.
networkState	네트워크 접속 상태를 표시한다.
readyState	미디어 파일의 다운로드 상태를 표시한다.

#### 1.6.2 error 속성의 값

속성	정수값	설명
MEDIA_ERR_ABORTED	1	미디어 파일의 다운로드가 사용자에게 의해 중지
MEDIA_ERR_NETWORK	2	네트워크 오류로 다운로드가 중지
MEDIA_ERR_DECODE	3	미디어 파일의 디코딩 실패
MEDIA_ERR_SRC_NOT_SUPPORTED	4	지원하지 않는 미디어 형식

1.6.3 networkState 속성의 값

속성	정수값	설명
NETWORK_EMPTY	0	요소의 초기 상태 표시
NETWORK_IDLE	1	네트워크를 실제 사용하고 있지 않으며, 네트워크에 접속하기 바로 직전 상태
NETWORK_LOADING	2	데이터의 다운로드 시도를 활발히 하고 있는 상태
NETWORK_NO_SOURCE	3	요소가 사용할 자원을 아직 찾지 못했다는 것을 의미

1.6.4 readyState 속성의 값

속성	정수값	설명
HAVE_NOTHING	0	다운로드 하지 않아 미디어 자원에 대한 정보가 없는 상태
HAVE_METADATA	1	메타 데이터 다운로드가 완료된 상태
HAVE_CURRENT_DATA	2	현재 재생 위치만큼 다운로드가 완료된 상태
HAVE_FUTURE_DATA	3	현재 재생 위치에 더해서 조금 더 재생할 수 있는 만큼 다운로드가 완료된 상태
HAVE_ENOUGH_DATA	4	HAVE_FUTURE_DATA + (현재의 재생 위치가 이용 가능한 데이터를 추월하지 않을 것으로 추정 or 브라우저가 더 이상 추가적으로 획득하는 데이터가 발생하지 않는다는 상태)

1.6.5 이벤트 처리

- 미디어 관련 주요 이벤트들은 다음과 같다.

```

play : 미디어 파일의 재생이 시작되었을 때 발생
timeupdate : 재생 중에 지속적으로 발생
waiting : 다음 프레임의 다운로드를 대기할 때 발생
ended : 재생이 종료되었을 때 발생
progress : 미디어 파일을 로드 할 때 지속적으로 발생
loadstart : 미디어 파일 로드를 시작할 때 발생
load : 미디어 파일의 로드가 완료될 때 발생
error : 미디어 파일을 재생/로드 할 때 오류가 있을 경우 발생
    
```

1.6.6 이벤트 등록 및 처리

- addEventListener() 메서드

```

var 변수 = document.getElementById('아이디');
//처리할 이벤트 등록
변수.addEventListener("이벤트", 실행할_함수, false);
...
//이벤트에 따른 처리 함수
function 실행할_함수 (e)
{
    //이벤트가 발생하였을 경우에 대한 처리
}

```

### 예제 오디오 제어

```

<!DOCTYPE html> <html> <head> </head>
<body>
<audio id="player" src="song.mp3" controls> </audio> <br>
<p>재생 상태 : <span id="playtime"> </span> </p>
<button onclick="play()">재생/일시정지 </button>
<button onclick="document.getElementById('player').volume+=0.1">볼륨 + </button>
<button onclick="document.getElementById('player').volume -= 0.1">볼륨 - </button>
<script>
    var player = document.getElementById('player');
    player.addEventListener("timeupdate", PlayTime, false);
    player.addEventListener("ended", PlayEnd, false);
    function play() {
        if(player.paused) player.play();
        else player.pause();
    }
    function PlayTime(e) {
        document.getElementById("playtime").innerHTML
        = Math.floor(player.currentTime) + "/" + Math.floor(player.duration);
    }
    function PlayEnd(e) {
        document.getElementById("playtime").innerHTML = "재생이 종료되었습니다.";
    }
</script>
</body> </html>

```



## 2 비디오

### 2.1 video 요소

- HTML5는 video요소를 정의하여 웹 페이지 안에서 비디오/무비를 내장하도록 지원해 준다.

```
<video src="비디오 파일 주소">
지원 불가능한 브라우저에서 표현할 내용
</video>
```



```
<video src="video.mp4" width="400px" controls autoplay loop>
현재의 브라우저는 video 요소를 지원하지 않습니다.
( <embed src="video.mp4"></embed> )
</video>
```

### 2.2 브라우저에서 지원하는 비디오 포맷

	IE 11	Firefox 32	Chrome 37	Safari 8.0	Opera 24
ogg	X	○	○	X	○
mp4	○	X	○	○	X
WebM	X	○	○	X	○

### 2.3 source 요소의 사용과 코덱 지정

- 브라우저별로 지원하고 있는 비디오 포맷이 다르기 때문에 다음과 같이 source 요소를 사용할 수 있다.

```
<video controls>
<!-- mp4 파일 안에 H.264 형식의 영상과 AAC 형식의 음성이 함께 있을 때 -->
<source src="video.mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2" '>
<!-- ogg 파일 안에 Theora 형식의 영상과 Vorbis 형식의 음성이 함께 있을 때 -->
<source src="video.ogg" type='video/ogg; codecs="theora, vorbis" '>
<embed src="video.mp4"></embed>
</video>
```

### 2.4 재생 시간 지정

- 미디어 콘텐츠를 재생할 때, 특정 위치를 지정하여 재생하는 것이 가능하다. video 요소의 src속성에 미디어의 URI를 지정할 때, 해시기호("#")와 함께 시작 시간과 종료 시

간을 지정하면 시작 시간부터 재생하기 시작하여 종료시간까지 재생을 한다.

#t=[시작시간], 종료시간]	#t=시간:분:초 [,시간:분:초]
src="video.mp4#t=1:20:32, 2:05:00"	1시간20분32초~2시간5분까지 재생
src="video.mp4#t=10, 20"	10초~20초까지 재생
src="video.mp4#t=,10"	10초까지 재생
src="video.mp4#t=20"	20초부터 재생

## 2.5 자막 처리

- track 요소는 여러 언어나 비디오 설명을 제공하거나 시간이 지정된 텍스트 파일을 지정할 때 사용한다.

### 2.5.1 track 요소의 사용

- 미디어 요소(audio, video)의 하위 요소로 track 요소를 사용하면 된다. 지정된 텍스트 파일 안에 이는 자막은 기본적으로 플레이어 아래 쪽에 표시된다.

```
<video id="player" src="video_Wildlife.mp4" controls autoplay loop>
  <track src='video_caption.vtt' srclang='ko' label='한국어' kind='subtitles' default>
</video>
```

### 2.5.2 트랙 파일 형식

- 텍스트 트랙은 WebVTT(Web Video Text Tracks)의 시간이 지정된 텍스트 파일 형식을 사용하는데, 이 파일은 UTF-8 형식의 텍스트 파일이다. WebVTT에 대한 자세한 내용은 표준화 문서 (<http://dev.w3.org/html5/webvtt/>)를 참조한다.

```
WEBVTT
00:00:02.000 --> 00:00:05.031
지금 보고 계시는 것은 예제로 만든 자막입니다.

00:00:07.000 --> 00:00:15.000
이런 식으로 자막을 비디오에 넣을 수 있습니다.

00:00:10.000 --> 00:20:15.000
생각보다 쉽게 자막을 처리할 수 있습니다.
```

### 2.5.3 다중 트랙 파일

- 여러 트랙을 사용하는 경우 한 트랙을 사용자가 언어를 지정 하지 않거나 선택하지 않은 경우에 사용할 기본 트랙으로 설정한다.

```
<video id="myVideo" controls autoplay loop>
  <source src="video.mp4" type="video/mp4">
  <track id="track_kr" src='video_ko.vtt' srclang='ko' label='한국어' kind='subtitles'
  default>
  <track id="track_en" src='video_en.vtt' srclang='en' label='영어' kind='subtitles' >
  <track id="track_de" src='video_de.vtt' srclang='de' label='독일어' kind='subtitles'>
</video>
```

**예제** 트랙을 이용한 비디오 재생

```
<!DOCTYPE html>
<html>
<head>
<title>비디오 재생하기</title>
</head>
<body>
<video id="player" src="video.mp4" width="500px" controls autoplay loop>
<track src='video_caption.vtt' srclang='ko' label='한국어' kind='subtitles' default >
</video>
</body>
</html>
```

