

# 웹 표준에 맞는 HTML5 프로그래밍 강의 노트

## 제 26회차

HTML5 API 종류 : Canvas, Video & Audio,  
Drag and Drop, Geolocation API

### ■ 학습목표

- HTML5의 API에 대해 설명할 수 있다.
- API를 활용할 수 있다.

### ■ 학습내용

- Canvas와 Video & Audio
- Drag and Drop과 Geolocation API

## 1. Canvas와 Video & Audio

### 1) Canvas

#### - Canvas

- HTML5가 나오기 전부터 존재함
- HTML5의 공식 명세로 자리잡음
- 2차원의 비트맵 이미지 프로세싱이 가능함
- 동적인 그래픽 렌더링을 스크립트로 제어할 수 있음

#### - Canvas 사용 방법

- Canvas는 넓이와 높이가 고정되어 있으므로 width와 height 속성을 사용해서 지정함
- 두 속성은 JavaScript를 이용하여 속성값을 변경할 수도 있음
- 속성값이 다시 지정되면 Canvas 내용은 모두 초기화 됨

〈넓이 100px, 높이 50px의 배경이 투명한 캔버스〉

```
<canvas id="ex" width="100" height="50"></canvas>
```

〈지정된 캔버스 위에 JavaScript를 사용하여 그림을 그릴 경우〉

```
var canvas = document.getElementById("ex");
var context = canvas.getContext("2d");
```

#### - context.fillText 메소드 : 캔버스에 문장을 작성할 때, 속을 채워서 그리는 방식

```
context.fillText("문자열", x, y);
```

#### - context.fillRect 메소드 : 사각형을 그리는 메소드

```
context.fillRect(x, y, width, height);
```

#### - context.beginPath 메소드 : 선 그리기의 시작을 나타내는 메소드

```
context.beginPath();
```

## 1. Canvas와 Video & Audio

### 1) Canvas

- context.moveTo 메소드 : 붓의 시작위치를 변경하는 메소드

```
context.moveTo(x, y);
```

- context.lineTo 메소드 : 붓의 끝 위치를 변경하는 메소드

```
context.lineTo(x, y);
```

- context.closePath 메소드 : 다 그리지 못한 부분을 선으로 이을 때 사용

```
context.closePath();
```

- 삼각형 그리기 실습 예제

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Triangle</title>
<style type="text/css">
        #myCanvas{
                                border:1px solid red;
        }
</style>
<script type="text/javascript">
        function start(){
        var ctx = document.getElementById("myCanvas").getContext("2d");
        ctx.beginPath();
        ctx.moveTo(50,50);
        ctx.lineTo(200,50);
        ctx.lineTo(50,200);
        //context.fill();
        ctx.closePath();
        ctx.stroke();
        }
</script>
</head>
<body onload="start();">

        <canvas id="myCanvas" width="300px" height="400px"></canvas>

</body>
</html>
```

## 1. Canvas와 Video & Audio

### 1) Canvas

- 사각형 그리기 실습 예제

```
<!DOCTYPE html>
<html>
<head>
<title>ex02</title>
    <script>
        window.onload = function () {
            var canvas = document.getElementById("myCanvas");
            var context = canvas.getContext("2d");

            //1번(사각형)
            context.rect(10, 30, 80, 100);
            context.stroke();
            //2번(선이있는사각형)
            context.strokeRect(100, 20, 150, 100);

            //3번(채워진사각형)
            context.fillRect(270,40,100,80);
        };
    </script>
</head>
<body>
    <canvas id="myCanvas" width="700" height="400">
    </canvas>
</body>
</html>
```

## 1. Canvas와 Video & Audio

### 2) Video & Audio

#### - Video & Audio의 특징

- HTML5에서 지원하는 새로운 요소
- 별도의 브라우저 플러그인을 이용하지 않고 웹 페이지에 쉽게 추가함
- 스크립트를 이용하여 미디어를 직접 제어함
- 플래시 등과 같은 플러그인을 이용하여 미디어를 재생할 수 있기 때문임
- <canvas> 요소를 결합하면 영상의 실시간 비트맵 연산이 가능함
- 다양한 그래픽 효과를 부여하거나 일종의 영상 판독기와 같은 애플리케이션을 오픈 웹 기술만으로 개발할 수 있음

#### - 기본 사용법

```
<video src="http://www.youtube.com/watch?v=iyafB0VCLsA"></video>
<audio src="http://www.youtube.com/watch?v=Dwit11VE16E"></audio>
```

#### - Video 실습 예제

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>ex03 - video</title>
</head>

<body>

        <video src="http://media.w3.org/2010/05/sintel/trailer.mp4"
width="640" height="480" autoplay controls>
        </video>

</body>
</html>
```

## 1. Canvas와 Video & Audio

### 2) Video & Audio

#### - Audio 실습 예제

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>ex04 - Audio</title>
</head>

<body>

    <audio src="koto-and-shamisen-loop-2.mp3" autoplay controls loop>
    Your browser does not support the audio element.
</audio>

</body>
</html>
```

## 2. Drag and Drop과 Geolocation API

### 1) Drag and Drop

- Drag and Drop API를 지원하는 HTML5

- 더욱 향상된 끌어다 놓기 경험을 제공함
- File API를 통해 바탕화면 혹은 탐색기의 파일을 브라우저로 직접 끌어다 놓는 방식으로도 파일을 업로드 할 수 있게 됨

- Draggable 요소

- 드래그 대상이 웹 애플리케이션 안의 요소라면 그 요소에 draggable 속성을 지정함
- Draggable 속성이 가질 수 있는 값

```
<li id="ex" draggable="true"> Example </li>
```

- Event

이벤트	이벤트를 알리는 곳	설명
dragstart	드래그 대상 요소	드래그가 시작됨
drag	드래그 대상 요소	드래그 중
dragenter	드래그 중 마우스 커서가 위치한 요소	드래그 조작이 요소 안의 범위에 들어옴
dragover	드래그 중 마우스 커서가 위치한 요소	드래그 조작이 요소 안의 범위를 통과 중
dragleave	드래그 중 마우스 커서가 위치한 요소	드래그 조작이 요소 안의 범위를 벗어남
drop	드롭 할 곳의 요소	드롭 되었음
dragend	드래그 대상 요소	드래그 종료



## 2. Drag and Drop과 Geolocation API

### 1) Drag and Drop

- ※ Drag and Drop 실습 예제

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>ex05 - Drag and Drip</title>
    <style>
    #div1 {
    width: 450px;
    height: 100px;
    padding: 10px;
    border: 1px dotted red;
    }
</style>
<script>

    function allowDrop(ev) {
    ev.preventDefault(); //요소위에 다른요소가 올수 있도록 허용
    }

    function drag(ev) {
    ev.dataTransfer.setData("text", ev.target.id); //드래그할때 클릭된 img의
text형식의 id값을 가져간다.
    }

    function drop(ev) {
    ev.preventDefault(); //요소위에 다른요소가 올수 있도록 허용
    var id = ev.dataTransfer.getData("text"); //클릭된 대상의 img id를 받음
    ev.target.appendChild(document.getElementById(id)); //클릭된 id의 img
를 div자식요소로 적용
    }

</script>
</head>
<body>

    <p> 원하는 이미지를 끌어서 옮기세요.</p>

    <div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>
    <br>
    
    
    
</body>
</html>

```

## 2. Drag and Drop과 Geolocation API

### 2) Geolocation API

- Geolocation API : 브라우저가 사용자의 지리적 위치를 찾아내고 그 정보를 어플리케이션에서 이용할 수 있도록 하는 기능
- 사용자의 위치 정보를 이용하기 위한 방법

- 사용자 콘텐츠가 생성될 때 지오-태깅(geo-tagging) 기능을 제공할 수 있음
- 근처에서 촬영된 사진 등에 대한 정보를 유기적으로 연결시켜 서비스 함
- 사용자의 위치가 변경될 때마다 콜백 메소드로 전달되어 항상 최신의 위치정보를 유지하는 것이 가능함
- 지리정보는 GPS 장치로부터 얻어지는 것이 가장 정확함

- Geolocation 사용법 : 현 위치 정보를 확인하기 위한 메소드

`getCurrentPosition(successCallback, errorCallback, options)`

1. 현재 위치 정보를 비동기로 확인
2. 처리 결과를 인수로 하여 successCallback을 한 번만 호출
3. 상세한 에러 정보를 인수로 하여 errorCallback을 호출함

#### 〈Geolocation 옵션〉

속성/상수	설명
<code>enableHighAccuracy</code>	<ul style="list-style-type: none"> <li>• 정확도가 높은 위치 정보를 요청함</li> </ul>
<code>timeout</code>	<ul style="list-style-type: none"> <li>• 위치정보 확인에 시간제한을 설정함(밀리세컨드)</li> <li>• 시간제한을 초과하면 에러(TIMEOUT)가 발생함</li> </ul>
<code>maximumAge</code>	<ul style="list-style-type: none"> <li>• 위치정보의 유효 기간을 설정함(밀리 세컨드)</li> <li>• 현재의 위치 정보가 <code>maximumAge</code>보다 오래된 것이라면 해당 위치 정보는 폐기되고 새롭게 위치 정보 확인을 시도함</li> <li>• 0을 지정하면 항상 새로운 위치 정보를 확인함</li> </ul>

## 2. Drag and Drop과 Geolocation API

### 2) Geolocation API

#### - Geolocation 실습 예제

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>ex06 - Geolocation</title>
</head>

<body onLoad="getLocation()">
  <p id="demo"></p>
  <script>
    var x=document.getElementById("demo");
    function getLocation() {
      if (navigator.geolocation) {

        navigator.geolocation.getCurrentPosition(showPosition);
      }
      else{
        x.innerHTML="이 브라우저는 geolocation을 지원하지 않습니다.";
      }
      function showPosition(position) {
        x.innerHTML="Latitude: " + position.coords.latitude + "<br />Longitude: " + position.coords.longitude;
      }
    }
  </script>
</body>
</html>
```

## ▣ 정리하기

### 1. Canvas와 video & Audio

- **<Canvas>** 요소 : 2차원의 비트맵 이미지 프로세싱이 가능하고 동적인 그래픽 렌더링을 스크립트로 제어할 수 있음
  - Canvas는 넓이와 높이가 고정되어 있으므로 **width**와 **height** 속성을 사용하면 됨
- **<video>**와 **<audio>** 요소를 이용하여 별도의 브라우저 플러그인을 이용하지 않고 미디어를 웹 페이지에 쉽게 추가할 수 있음

### 2. Drag and Drop과 Geolocation API

- 드래그 대상이 웹 애플리케이션 안의 요소라면 그 요소에 **draggable** 속성을 지정함
- Draggable 속성이 가질 수 있는 값 : **'true'**, **'false'**, **'값 없음'**
- **Geolocation API** : 브라우저가 사용자의 지리적 위치를 찾아내고 그 정보를 어플리케이션에서 이용할 수 있도록 하는 기능