|  |
| --- |
| **HTML5 Audio 기능을 활용한 Recoding 기능 구현하기** |

본 문서에서는 MediaDevices 인터페이스와 HTML5 audio 태그 스펙에 대한 간단한 소개와 WebSquare5에서 MediaDevices와 WebSquare audio 컴포넌트(HTML5 audio 태그 매핑 컴포넌트)를 이용한 Recoding 기능을 구현하는 방법을 설명합니다.

# 기반 기술 소개

Recoding 기능 구현을 위해서 사용된 MediaDevices 인터페이스와 HTML5 audio 태그에 대해서 설명합니다.

## MediaDevices

MediaDevices 인터페이스는 카메라, 마이크, 화면공유 등 웹 브라우저 상에서 사용자의 디바이스(PC, 태블릿, 스마트폰 등)에 연결된 미디어 입력 장치에 접근하는 방법을 제공하는 인터페이스입니다.

### Event

|  |  |
| --- | --- |
| 이벤트명 | 설명 |
| devicechange | 사용자 디바이스에 미디어 입출력 장치가 추가되거나 제거될 때 발생합니다. |

### Method

|  |  |
| --- | --- |
| 메스드명 | 설명 |
| enumerateDevices() | 시스템에서 사용 가능한 미디어 입출력 장치의 정보를 담은 배열을 가져옵니다. |
| getSupportedConstraints() | 인터페이스가 지원하는 제약을 나타내는 MediaTrackSupportedConstraints 호환 객체를 반환합니다. |
| getDisplayMedia() | MediaStream으로 화면을 캡처해 공유나 녹화 용도로 사용할 전체 화면 혹은 화면 윈도우를 선택하도록 사용자에게 요청합니다. MediaStream으로 이행하는 Promise를 반환합니다. |
| getUserMedia() | 사용자에게 권한을 요청한 후, 시스템의 카메라와 오디오 각각 혹은 모두 활성화하여, 장치의 입력 데이터를 비디오/오디오 트랙으로 포함한 MediaStream을 반환합니다. |

### 웹 브라우저 별 MediaDevices 인터페이스 지원 현황

웹 브라우저 별 MediaDevices 인터페이스 지원 현황입니다. 모바일 웹 브라우저의 지원 현황을 확인하려면 <https://caniuse.com/#search=mediaDevices> 웹 페이지를 참고하시기 바랍니다. MediaDevices 인터페이스가 IE에서는 지원되지 않기 때문에, 본 가이드의 예제는 IE 환경을 지원하지 않습니다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **IE** | **Edge** | **Firefox** | **Chrome** | **Safari** | **Opera** |
| 미지원 | 18 | 33 | 47 | 11 | 30 |

※ 숫자는 웹 브라우저 별 MediaDevices 인터페이스를 지원하는 최소 버전임

MediaDevices 인터페이스에 접근하기 위해서는 웹 브라우저 별 지원 제약사항 이외에도, 웹 페이지가 Secure Context(HTTPS/TLS)로 로딩되어야 하는 추가적인 제약사항이 있습니다. 개발 과정에서 테스트를 위한 로컬 호스트(127.0.0.1, localhost)로 접근 시에는 웹 페이지가 Secure Context(HTTPS/TLS)로 로딩되지 않아도 MediaDevices 인터페이스에 접근합니다. 하지만, IP나 도메인으로 웹 페이지에 접근한다면, Secure Context (HTTPS/TLS)로 웹 페이지를 로딩해야 합니다. Secure Context (HTTPS/TLS)아 아닌 상태로 웹 페이지에 접근할 경우 “TypeError: Cannot read property 'getUserMedia' of undefined”라는 에러 메시지가 출력되며 에러가 발생합니다.

## HTML5 <audio> Tag

HTML5 이전에는 웹 페이지에서 Audio를 플레이하기 위한 표준안이 없었기 때문에, 웹 페이지 상에서 Audio나 Video를 포함시키기 위해서 Flash나 ActiveX와 같은 비표준 기술을 사용해야 했습니다 하지만, HTML5 스펙에는 Audio와 Video를 플레이 하기 위한 표준이 추가되었고, 현재 대부분의 범용 브라우저에서는 HTML5의 audio와 video 태그를 지원하고 있습니다.

### audio 태그 속성

audio 태그에서 지원하는 속성은 다음과 같습니다.

|  |  |
| --- | --- |
| 속성명 | 설명 |
| autoplay | audio 파일의 실행 준비가 완료되는 대로 소리 재생을 시작하도록 설정함  전체 audio 파일이 다운로드 시점까지 기다리지 않고 실행할 수 있는 가능 빠른 시점에 재생을 시작함 |
| controls | Audio 재생, 볼륨, 탐색, 일시정지 버튼과 같이 오디오의 실행을 제어할 수 있는 버튼의 디스플레이 상태를 설정함 |
| loop | audio 재생이 끝나면 자동으로 다시 재생시킬 것인지 설정함 |
| muted | audio의 소리 출력이 음소거 되도록 설정함 |
| preload | 웹 페이지가 로드 될 때 audio 파일이 같이 로드되어야 하는지 여부와 그 방법을 설정함   |  |  | | --- | --- | | 속성 명 | 설명 | | auto | 웹 페이지가 로드 될 때, 웹 브라우저가 전체 audio 파일을 같이 로딩 하도록 설정함 | | metadata | 웹 페이지가 로드 될 때, 웹 브라우저가 Meta Data만 로딩 하도록 설정함 | | none | 웹 페이지가 로드 될 때, 웹 브라우저가 audio 파일을 로딩 하지 않도록 설정함 | |
| src | audio 파일의 URL을 설정함 |

본 예제에서는 MediaDevices 인터페이스를 통해서 입력된 소리 데이터를 audio 태그를 이용해서 플레이하고 있습니다. 별도의 소리 데이터 파일(mp3, wav, ogg)을 audio 태그를 이용해서 플레이하려면, source 태그를 이용해서 접근할 파일을 설정하면 됩니다.

### 웹 브라우저 별 audio Tag 지원 현황

아래는 웹 브라우저 별 audio Tag 지원 현황 입니다. 모바일 웹 브라우저 별 지원 현황을 확인하려면 <https://caniuse.com/#feat=audio> 웹 페이지를 참고하시기 바랍니다.

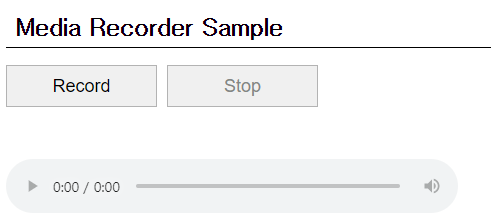
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **IE** | **Edge** | **Firefox** | **Chrome** | **Safari** | **Opera** |
| 9 | 12 | 3.5 (부분지원)  20 (전체지원) | 4 | 4 | 10.1 (부분지원)  11.5 (전체지원) |

※ 숫자는 웹 브라우저 별 audio 태그를 지원하는 최소 버전임

# WebSquare5에서 Recoding 기능 개발하기

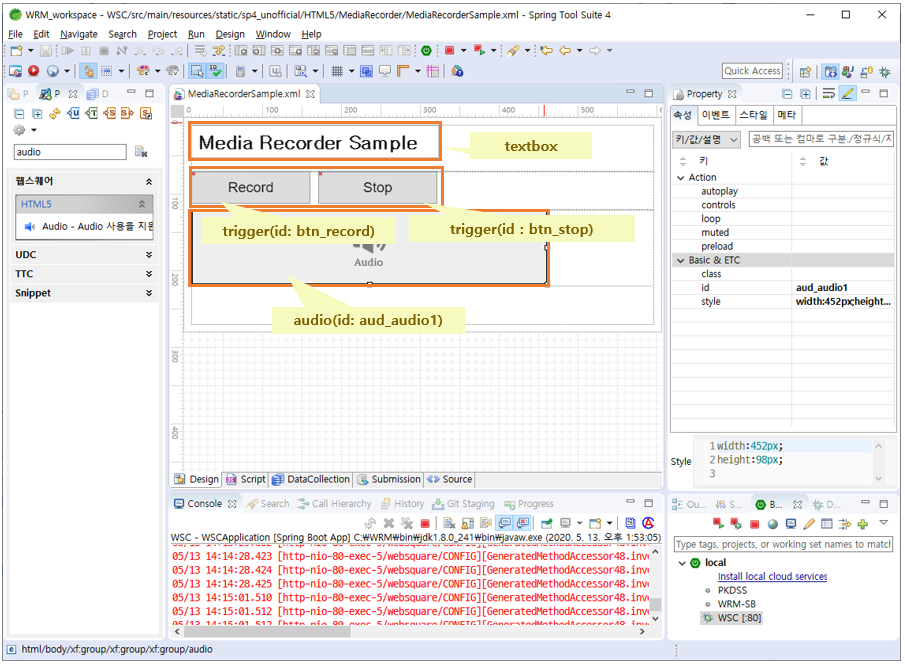
WebSquare5에서 MediaDevices 인터페이스와 HTML5 audio 태그 사용해서, 아래의 화면과 같이 소리를 녹음하고 플레이하는 간단한 Recoding 기능을 제공하는 화면을 개발해보도록 하겠습니다.

IE에서는 MediaDevices 인터페이스를 지원하지 않기 때문에 본 예제는 동작하지 않습니다.



## WebSquare5 Studio를 이용해서 화면 그리기

WebSquare5 Studio를 이용해서 아래와 같이 화면 레이아웃을 구성합니다.



## Recoding 기능 구현을 위한 Script 작성

웹 브라우저에서 WebSquare 화면을 로딩 시, 화면이 로딩된 후 자동으로 실행되는 onpageload 이벤트 함수에서 mediaDevices 인터페이스 접근해서 MediaRecorder 객체를 이용해서 녹음을 시작할 수 있도록 세팅하는 코드를 작성합니다.

|  |
| --- |
| scwin.onpageload = **function**() {  **if** (**typeof** navigator.mediaDevices !== "undefined") {  navigator.mediaDevices.getUserMedia({  audio : **true** // Audio 기능만 사용하도록 audio 속성만 true로 설정함  }).then(**function**(stream) {  // Audio Device가 정상적으로 로딩될 경우  btn\_record.setDisabled(**false**); // Record 버튼을 활성화  btn\_stop.setDisabled(**true**); // Stop 버튼을 비활성화    **var** option = {  audioBitsPerSecond : 128000, // Audio 인코딩 비트 전송률  mimeType : "audio/webm" // Audio 파일 MIME TYPE  };    // Audio 데이터 녹음을 위한 MediaRecorder 객체 생성  scwin.recorder = **new** MediaRecorder(stream, option);  // dataavailable 이벤트 설정  scwin.recorder.addEventListener("dataavailable", scwin.onRecordingReady);  }).**catch**(**function**(ex) {  // Audio Device가 로딩 과정에서 에러 발생  console.error(ex);  });  } **else** {  console.error("navigator.mediaDevices에 접근할 수 없습니다.");  }  }; |

Record 버튼 클릭 시, Stop 버튼을 활성화 시키고 Record 버튼은 중복해서 클릭하지 못하도록 비활성화 시킵니다. 그리고, scwin.recorder(MediaRecorder) 객체의 start() 메소드를 실행시켜서 녹음을 시작합니다. 녹음은 사용자가 Stop 버튼을 클릭할 때까지 계속 진행됩니다.

|  |
| --- |
| /\*\*  \* Record 버튼 클릭 시  \*/  scwin.btn\_record\_onclick = **function**(e) {  btn\_record.setDisabled(**true**);  btn\_stop.setDisabled(**false**);  scwin.recorder.start(); // MediaRecorder 객체를 이용한 녹음 시작  }; |

Stop 버튼을 클릭 시, Record 버튼을 다시 활성화 시키고, Stop 버튼을 다시 클릭할 수 없도록 비활성화 시킵니다. 그 후에 scwin.recorder(MediaRecorder) 객체의 stop() 메소드를 실행시켜서 녹음을 종료합니다. 녹음이 종료되면 scwin.recorder(MediaRecorder) 객체에 바인딩된 dataavailable 이벤트가 실행됩니다.

|  |
| --- |
| /\*\*  \* Stop 버튼 클릭 시  \*/  scwin.btn\_stop\_onclick = **function**(e) {  btn\_record.setDisabled(**false**);  btn\_stop.setDisabled(**true**);  scwin.recorder.stop(); // MediaRecorder 객체를 이용한 녹음 종료  }; |

녹음 종료 시 MediaRecorder dataavailable 이벤트가 발생합니다. 이 이벤트 함수에서 URL.createObjectURL() 메소드를 호출해서 이벤트 데이터(e.data)를 녹음된 데이터 객체를 가리키는 URL로 생성합니다. 이 URL을 Audio 컴포넌트의 src 속성에 저장한 후, 녹음된 데이터를 플레이하기 위해서 play() 메소드를 실행합니다.

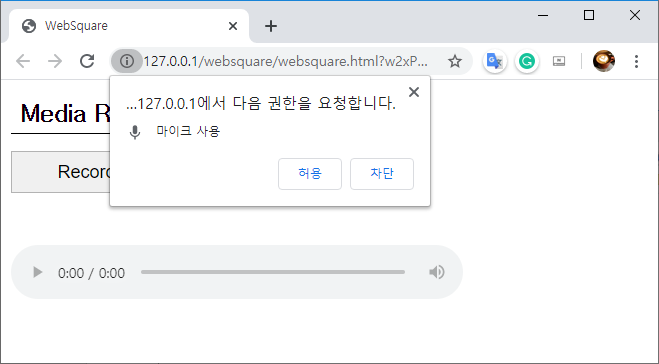
|  |
| --- |
| /\*\*  \* dataavailable 이벤트 발생 시 (녹음 종료 시 발생함)  \*/  scwin.onRecordingReady = **function**(e) {  // 녹음된 데이터 객체를 가리키는 URL을 생성해서 Audio 객체의 src에 세팅함  aud\_audio1.src = URL.createObjectURL(e.data);  aud\_audio1.play();  }; |

화면이 닫힐 때는 메모리 누수를 방지하기 위해서 URL.revokeObjectURL() 메소드를 이용해서 녹음된 데이터 객체를 가리키는 URL을 해제해야 합니다.

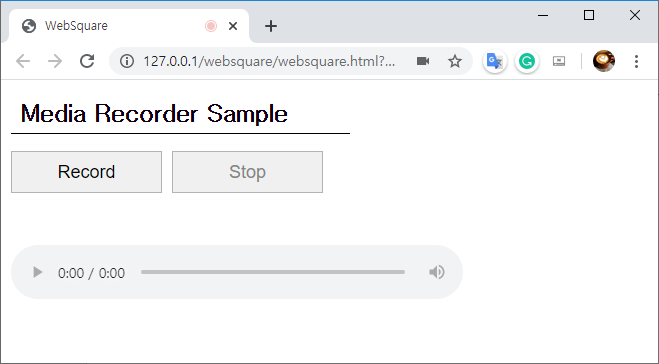
|  |
| --- |
| scwin.onpageunload = **function**() {  // 페이지가 unload될 때 녹음된 데이터 객체를 가리키는 URL 해제함  URL.revokeObjectURL(aud\_audio1.src);  }; |

## Media Recorder 예제 실행하기

작성된 예제 프로그램을 최초 실행하게 되면, 마이크 사용 권한이 없는 사이트의 경우 아래와 같이 ‘마이크 사용’ 권한 허용을 요청하는 확인창이 오픈 됩니다. ‘허용’ 버튼을 클릭합니다.



마이크 허용된 웹 페이지에 접속 시, Chrome Browser의 경우 탭의 좌측에 빨간색 원형 아이콘이 나타납니다.



Record 버튼을 클릭하면 녹음이 시작되고, Stop 버튼을 클릭하면 녹음이 종료되면서, Audio 컴포넌트를 통해서 녹음된 소리가 자동으로 플레이 됩니다.

# 참고 사이트

* <https://developer.mozilla.org/en/docs/Web/API/MediaDevices>
* <https://developer.mozilla.org/en/docs/Web/API/MediaDevices/getUserMedia>
* <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/audio>
* <https://developer.mozilla.org/en/docs/Web/API/URL/createObjectURL>
* <https://developer.mozilla.org/en/docs/Web/API/URL/revokeObjectURL>
* <https://caniuse.com/#search=mediaDevices>
* <https://caniuse.com/#feat=audio>