

Kollus 전용플레이어 Javascript Library

작성일	문서버전	비고
2014.11.25	v1.0	문서 초안 작성
2014.12.01	v0.3.3	loaded 이벤트 추가, set_repeat 메소드 (구간반복설정) 추가 * 문서 버전을 실제 스크립트의 버전과 일치되도록 수정되었습니다. 스크립트가 버전업 되더라도 문서를 수정할 필요가 없는 경우는 현재 문서버전을 그대로 유지합니다.
2014.12.02	v0.3.4	set_repeat 메소드에 구간반복 커스텀 설정 기능 추가
2014.12.04	v0.3.5	set_playlist_loop 메소드 추가
2015.01.30	v0.3.7	<ul style="list-style-type: none"> 초기 설정에 if_lower_version 추가 기존 if_not_installed 설정이 IE 뿐만 아니라 크롬 계열에서도 동작하도록 수정
2015.05.18	v.0.4.0	error 이벤트 리스너 추가

목차

[목차](#)

[주요 기능](#)

[유의 사항](#)

[재생 옵션 목록](#)

[지원 이벤트 목록](#)

[지원 메소드 목록](#)

[Library 설치 샘플](#)

주요 기능

- Kollus 비디오 게이트웨이를 이용하지 않고도 직접 고객사의 웹페이지에 Kollus 전용플레이어(PC)를 장착하여 서비스할 수 있습니다.
- 플레이리스트를 지원합니다.
- Kollus 전용 플레이어(PC) 자체에서 제공하는 대부분의 기능을 javascript로 컨트롤할 수 있습니다.

유의 사항

- 본 javascript library는 PC용 Kollus 전용플레이어만 지원합니다.
- 자동 업데이트를 지원하지 않으므로 Kollus 전용플레이어 버전 업데이트시에 직접 업데이트된 버전을 library 로딩 옵션에 추가해야 합니다.
- Kollus 전용플레이어의 인스톨 파일을 직접 서비스하고자 한다면 파일명은 그대로 유지해야 합니다. (도메인과 경로는 고객사의 것으로 대체해도 무방합니다.)
- 플레이리스트 기능 이용시에 각 URL의 만료시간은 모든 동영상이 재생될때까지 유지되도록 충분히 길게 설정해야 합니다.
- 플레이리스트 기능 이용시에는 반드시 하나 이상의 콘텐츠가 playlist 옵션에 설정되어 있어야 합니다.
- 플레이리스트 기능 이용시에 remove_url_from_playlist 메소드를 호출하여 url을 playlist로부터 제거한 뒤에는 playlist에 적어도 콘텐츠가 하나 이상은 남아 있어야 합니다. playlist 콘텐츠를 모두 제거한 경우는 의도된대로 플레이리스트가 재생되지 않습니다.

재생 옵션 목록

옵션 이름	Data type	필수 여부	설명
placeholder	string	필수	플레이어 embed 코드가 삽입될 부모 HTML Tag의 ID입니다.
width	string	선택	플레이어의 가로 크기 입니다. (기본값 720px)
height	string	선택	플레이어의 세로 크기 입니다. (기본값 480px)
skinurl	string	선택	커스텀 스킨 사용시 사용할 스킨의 URL 경로입니다.
skinsha1sum	string	선택	커스텀 스킨 사용시 사용할 스킨의 sha1sum값입니다.
mediainfourl	string	필수	재생할 콘텐츠의 재생정보 URL입니다. (이 값과 playlist값이 동시에 설정되면 playlist 설정이 우선되고 mediainfourl값은 무시됩니다.)
playlist	array	선택	플레이리스트 기능 사용시 재생할 콘텐츠의 URL을 배열로 지정합니다. 예> ['http://...', 'http://...']
playlist_loop	boolean	선택	플레이리스트를 반복하여 재생할지 여부를 설정합니다. (기본값 false)
player_version	integer	필수	사용중인 플레이어의 버전입니다. 예> 2006
installation_url	string	필수	플레이어 설치파일의 경로입니다.

if_not_installed	function	선택	플레이어가 설치되어있지 않은 경우 callback으로 실행될 함수입니다.
if_lower_version	function	선택	설치된 플레이어가 player_version보다 낮은 경우 callback으로 실행될 함수입니다.

지원 이벤트 목록

* 이벤트 리스너를 등록하는 방법은 하단의 Library 설치 샘플 섹션을 참고하여 주십시오.

이벤트 이름	이벤트 발생 시점	설명
progress	재생시 매초 발생	<ul style="list-style-type: none"> • 재생시 매초마다 발생합니다. • params <ul style="list-style-type: none"> ○ (int) percent 재생된 분량의 백분율. 값의 범위는 $0 \leq \text{percent} \leq 100$ ○ (int) position 현재 재생중인 위치. 초단위. ○ (int) duration 컨텐츠의 전체 재생 분량. 초단위. (IE는 이 값이 int형태지만, non-IE 환경에서는 double 형태입니다.)
screenchange	화면 모드 변경시	<ul style="list-style-type: none"> • 전체, 일반화면으로 변경시 발생합니다. • params <ul style="list-style-type: none"> ○ (string) screen <ul style="list-style-type: none"> ■ 일반화면 -> 'windowed' ■ 전체화면 -> 'fullscreen'
speedchange	배속 변경시	<ul style="list-style-type: none"> • 배속 변경시 발생합니다. • params <ul style="list-style-type: none"> ○ (double) speed 변경된 배속 값. 값의 범위는 $0.5 \leq \text{speed} \leq 2$
volumechange	음량 변경시	<ul style="list-style-type: none"> • 음량 변경시 발생합니다. • params <ul style="list-style-type: none"> ○ (int) volume

		<p>변경된 음량 값. 값의 범위는 $0 \leq \text{volume} \leq 100$</p>
play	재생 시작시	<ul style="list-style-type: none"> ● 재생 시작시에 발생합니다. (초기 재생 시작시나 일시 멈춤 상태에서 다시 재생할때도 발생합니다.) ● params <ul style="list-style-type: none"> ○ no params
pause	일시 정지시	<ul style="list-style-type: none"> ● 일시 정지시에 발생합니다. ● params <ul style="list-style-type: none"> ○ no params
done	재생 완료시	<ul style="list-style-type: none"> ● 재생 완료시에 발생합니다. 재생 완료는 duration의 끝까지 모두 재생한 경우를 의미합니다. ● params <ul style="list-style-type: none"> ○ no params
ready	플레이어 로딩 완료시	<ul style="list-style-type: none"> ● 플레이어 로딩이 완료되면 호출됩니다. 플레이어가 실행되고 각종 설정 정보를 호출할 수 있는 시점입니다. 플레이리스트 기능 사용시에는 이 이벤트가 다음 순번의 콘텐츠를 재생할때도 발생합니다. ● params <ul style="list-style-type: none"> ○ no params
loaded	플레이어 재생 준비 완료시	<ul style="list-style-type: none"> ● 플레이어 로딩이 끝나고 재생 정보를 서버로부터 획득하였으며, 실제 재생준비가 완료된 시점입니다. ● params

		<ul style="list-style-type: none"> ○ no params
topmost	최상위 모드 설정시	<ul style="list-style-type: none"> ● 최상위 모드 변경시 발생합니다. ● params <ul style="list-style-type: none"> ○ (boolean) topmost <ul style="list-style-type: none"> ■ 최상위 모드 -> true ■ 일반 모드 -> false
videosettingchange	비디오 설정 변경시	<ul style="list-style-type: none"> ● 밝기, 대비, 색조 등의 값이 변경되는 시점에 발생합니다. (셋 중 하나라도 변경되면 발생합니다.) 각 값의 범위는 $-50 \leq \text{value} \leq 50$ 입니다. ● params <ul style="list-style-type: none"> ○ (int) brightness ○ (int) contrast ○ (int) saturation
jumpstepchange	이동 범위 변경시	<ul style="list-style-type: none"> ● 이동범위 변경시 발생합니다. ● params <ul style="list-style-type: none"> ○ (int) jumpstep 변경된 이동 범위 값
playlist	플레이리스트 아이템 재생시	<ul style="list-style-type: none"> ● 플레이리스트 아이템이 재생될때마다 호출됩니다. ● params <ul style="list-style-type: none"> ○ (string) url 재생할 url ○ (array) playlist 전체 playlist
error	플레이어 에러	<ul style="list-style-type: none"> ● 플레이어 에러 발생시 호출됩니다.

	발생시	<ul style="list-style-type: none">• params<ul style="list-style-type: none">○ (string) error_code
--	-----	---

지원 메소드 목록

메소드	설명
init(options, events)	<ul style="list-style-type: none"> 라이브러리 초기화 메소드 params <ul style="list-style-type: none"> (object) options 재생 옵션 목록 참조 (object) events 지원 이벤트 목록 참조 return <ul style="list-style-type: none"> no return
get_progress()	<ul style="list-style-type: none"> 현재 재생 진행상태 반환 params <ul style="list-style-type: none"> no params return <ul style="list-style-type: none"> (object) 진행상황 <ul style="list-style-type: none"> (int) percent 진행된 분량의 백분율 (0 <= percent <= 100) (int) position 현재 재생되고 있는 지점 (초) (int) duration 전체 재생시간 (IE의 경우 int형으로 반환되고 non-IE의 경우엔 double형으로 반환됩니다.)
play(start_at)	<ul style="list-style-type: none"> 재생 명령 params <ul style="list-style-type: none"> (int) start_at

	<p>특정 위치부터 재생하려면 해당 위치의 초를 int형으로 입력. (처음부터 재생하려면 start_at을 넣지 않고 호출합니다.)</p> <ul style="list-style-type: none"> • return <ul style="list-style-type: none"> ○ no return
pause()	<ul style="list-style-type: none"> • 일시 정지. 다시 재생하려면 play 메소드를 호출합니다. • params <ul style="list-style-type: none"> ○ no params • return <ul style="list-style-type: none"> ○ no return
set_playlist(playlist)	<ul style="list-style-type: none"> • 현재 설정된 playlist를 대체합니다. • params <ul style="list-style-type: none"> ○ (array) playlist 옵션의 playlist 항목과 같은 형태입니다. • return <ul style="list-style-type: none"> ○ no return
get_playlist()	<ul style="list-style-type: none"> • 현재 설정된 playlist를 반환합니다. • params <ul style="list-style-type: none"> ○ no params • return <ul style="list-style-type: none"> ○ (array) playlist
add_url_to_playlist(url)	<ul style="list-style-type: none"> • 현재 설정된 playlist에 url을 추가합니다. • params <ul style="list-style-type: none"> ○ (string) url 유의사항을 참고하여 주십시오. • return <ul style="list-style-type: none"> ○ no return

remove_url_from_playlist(url)	<ul style="list-style-type: none"> ● 현재 설정된 playlist에서 url을 제거합니다. ● params <ul style="list-style-type: none"> ○ (string) url 유의사항을 참고하여 주십시오. ● return <ul style="list-style-type: none"> ○ no return
playlist_next()	<ul style="list-style-type: none"> ● 플레이리스트로 재생시 다음 순번의 콘텐츠를 재생합니다. 마지막 순번이 재생중이면 메소드는 동작하지 않습니다. playlist_loop 옵션이 true인 경우엔 처음으로 되돌아갑니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ no return
playlist_prev()	<ul style="list-style-type: none"> ● 플레이리스트로 재생시 이전 순번의 콘텐츠를 재생합니다. 첫번째 순번의 콘텐츠가 재생중이면 메소드는 동작하지 않습니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ no return
volume(volume)	<ul style="list-style-type: none"> ● 플레이어의 음량을 제어합니다. ● params <ul style="list-style-type: none"> ○ (int) volume volume값의 범위는 $0 \leq \text{volume} \leq 100$입니다. 값이 범위를 벗어나면 메소드는 값을 무시합니다. ● return

	<ul style="list-style-type: none"> ○ no return
get_volume()	<ul style="list-style-type: none"> ● 현재 설정된 음량을 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ (int) 현재 설정된 볼륨
mute()	<ul style="list-style-type: none"> ● 음량을 일시적으로 0으로 설정합니다. 다시 호출하면 이전에 설정된 음량으로 회복됩니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ no return
speed(speed)	<ul style="list-style-type: none"> ● 배속을 설정합니다. ● params <ul style="list-style-type: none"> ○ (double) speed 배속 값의 범위는 $0.5 \leq \text{speed} \leq 2.0$ 입니다. 값이 범위를 벗어나면 메소드는 값을 무시합니다. ● return <ul style="list-style-type: none"> ○ no return
get_speed()	<ul style="list-style-type: none"> ● 설정된 배속값을 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ (double) 설정된 배속 값
ff()	<ul style="list-style-type: none"> ● 지정된 이동 범위(set_jump_step이나 플레이어의 재생->이동범위 선택 메뉴에 의해 설정된 값)만큼

	<p>앞으로 이동합니다.</p> <ul style="list-style-type: none"> • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ no return
rw()	<ul style="list-style-type: none"> • 지정된 이동 범위(set_jump_step이나 플레이어의 재생->이동 범위 선택 메뉴에 의해 설정된 값)만큼 뒤로 이동합니다. • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ no return
screen()	<ul style="list-style-type: none"> • 플레이어의 전체화면, 일반화면을 toggle합니다. • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ no return
get_screen()	<ul style="list-style-type: none"> • 현재 설정된 플레이어의 화면 모드를 반환합니다. • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ (string) 현재 설정된 플레이어의 화면 모드 <ul style="list-style-type: none"> ■ 일반 화면 -> 'windowed' ■ 전체 화면 -> 'fullscreen'
set_jump_step(step)	<ul style="list-style-type: none"> • 이동 범위를 설정합니다. 설정된 이동 범위는 ff, rw 메소드에서 사용합니다. • params <ul style="list-style-type: none"> ◦ (int) step

	<p>이동 범위. 초단위.</p> <ul style="list-style-type: none"> • return <ul style="list-style-type: none"> ◦ no return
get_jump_step()	<ul style="list-style-type: none"> • 현재 설정된 이동 범위를 반환합니다. • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ (int) 현재 설정된 이동 범위
top_most(topmost)	<ul style="list-style-type: none"> • 플레이어의 최상위 모드를 지정합니다. (다른 윈도우들은 플레이어 뒤로 배치됩니다.) • params <ul style="list-style-type: none"> ◦ (boolean) topmost <ul style="list-style-type: none"> ■ true -> 최상위 모드 지정 ■ false -> 최상위 모드 해제 • return <ul style="list-style-type: none"> ◦ no return
get_top_most()	<ul style="list-style-type: none"> • 현재 설정된 플레이어 최상위 모드 값을 반환합니다. • params <ul style="list-style-type: none"> ◦ no params • return <ul style="list-style-type: none"> ◦ (boolean) 최상위 모드 값
brightness(brightness)	<ul style="list-style-type: none"> • 플레이어의 밝기 값을 지정합니다. • params <ul style="list-style-type: none"> ◦ (int) brightness 밝기 값의 범위는 $-50 \leq \text{brightness} \leq 50$ 입니다. 범위를 넘어선 값은 무시됩니다. • return <ul style="list-style-type: none"> ◦ no return

get_brightness()	<ul style="list-style-type: none"> ● 현재 설정된 플레이어의 밝기 값을 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ (int) 플레이어 밝기 값
contrast(contrast)	<ul style="list-style-type: none"> ● 플레이어의 대비 값을 지정합니다. ● params <ul style="list-style-type: none"> ○ (int) contrast 대비 값의 범위는 $-50 \leq \text{contrast} \leq 50$ 입니다. 범위를 넘어선 값은 무시됩니다. ● return <ul style="list-style-type: none"> ○ no return
get_contrast()	<ul style="list-style-type: none"> ● 현재 설정된 플레이어의 대비 값을 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ (int) 플레이어 대비 값
saturation(saturation)	<ul style="list-style-type: none"> ● 플레이어의 색조 값을 지정합니다. ● params <ul style="list-style-type: none"> ○ (int) saturation 색조 값의 범위는 $-50 \leq \text{saturation} \leq 50$ 입니다. 범위를 넘어선 값은 무시됩니다. ● return <ul style="list-style-type: none"> ○ no return
get_saturation()	<ul style="list-style-type: none"> ● 현재 설정된 플레이어의 색조 값을 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return

	<ul style="list-style-type: none"> ○ (int) 플레이어 색조 값
set_control_visibility(visibility)	<ul style="list-style-type: none"> ● 플레이어 내부의 컨트롤 스킨의 표시 상태를 변경합니다. ● params <ul style="list-style-type: none"> ○ (boolean) visibility <ul style="list-style-type: none"> ■ true -> 화면 컨트롤을 표시하지 않음 ■ false -> 화면 컨트롤을 표시함 ● return <ul style="list-style-type: none"> ○ no return
get_control_visibility()	<ul style="list-style-type: none"> ● 플레이어의 컨트롤 스킨 표시 상태를 반환합니다. ● params <ul style="list-style-type: none"> ○ no params ● return <ul style="list-style-type: none"> ○ (boolean) 플레이어 컨트롤 스킨 표시 상태
set_repeat	<ul style="list-style-type: none"> ● 구간반복을 설정합니다. 처음 호출하면 현재 위치를 시작 지점으로, 다시 호출하면 현재 위치를 종료 지점으로 설정합니다. 만약에 아래 두가지 인자가 주어지면 지정한 위치로부터 구간반복을 설정합니다. ● params <ul style="list-style-type: none"> ○ (integer) kind (1 -> 시작, 2 -> 종료) ○ (integer) pos (초 단위, -1 입력시 현재 위치) ● return <ul style="list-style-type: none"> ○ no return
set_playlist_loop	<ul style="list-style-type: none"> ● 플레이리스트 종료시에 계속해서 반복할 것인지 여부를 설정합니다. (이 옵션은 초기 설정값에서도 지정할 수 있습니다.) ● params

	<ul style="list-style-type: none">○ (boolean) loop (true -> 반복, false -> 반복하지 않음)● return<ul style="list-style-type: none">○ no return
--	---

Library 설치 샘플

```

...
<head>
...
    // HTML의 head 태그 사이에 kollusplayer.min.js 스크립트 파일을 로딩하는 부분을
    // 추가합니다.
    <script src="kollusplayer.min.js"></script>

    // Kollusplayer library javascript는 로딩 후에 window 전역공간에 KollusPlayer 라는 재생
    // 제어 객체를 추가합니다. 플레이어 로딩이나 재생 제어는 모두 이 객체를 이용합니다.
    <script>
        window.onload = function() {
            KollusPlayer.init(
                // 재생과 관련한 options을 설정하는 부분입니다.
                // 각 항목에 대한 설명은 문서 상단의 '재생 옵션 목록'을 참조하여 주십시오.
                {
                    placeholder: 'Player',
                    width: '720px',
                    height: '480px',
                    skinurl: "",
                    skinsha1sum: "",
                    mediainfurl: 'http://v.kr.kollus.com.com/content1',
                    playlist: ['http://v.kr.kollus.com/content1', 'http://v.kr.kollus.com/content2',
                        'http://v.kr.kollus.com/content3'],
                    playlist_loop: true,
                    player_version: 2006,
                    installation_url: 'http://file.kr.dev.kollus.com/public/kollus2',
                    if_not_installed: function() {
                        alert('플레이어를 업데이트해야 합니다.');
```

```

        ...
    },
},
// 재생 이벤트 핸들러를 지정하는 부분입니다.
// 사용 가능한 이벤트는 문서 상단의 '지원 이벤트 목록'을 참조하여 주십시오.
{
    progress: function(percent, position, duration) {
        ...
    },
    screenchange: function(screen) {
        ...
    },
    speedchange: function(speed) {
        ...
    },
    volumechange: function(volume) {
        ...
    },
    play: function() {
        ...
    },
    pause: function() {
        ...
    },
    done: function() {
        ...
    },
    ready: function() {
        ...
    },
},

```

```
        topmost: function(topmost) {  
            ...  
        },  
        videosettingchange: function(brightness, contrast, saturation) {  
            ...  
        },  
        jumpstepchange: function(jumpstep) {  
            ...  
        },  
        error: function(error_code) {  
            ...  
        }  
    }  
};  
);  
}  
</script>  
...  
</head>  
<body>  
...  
    // 플레이어 embed 코드가 삽입될 placeholder HTML Tag 입니다.  
    <div id="Player"></div>  
...  
</body>  
...
```