

# SF.SERVICE.VIDEOPLAYER

**\*\* This class will not be supported in 2015.**

All functionalities of `sf.service.VideoPlayer` class are more improved, integrating with CAPH. Therefore `sf.service.VideoPlayer` class is not supported since 2015 Smart TV. To use functions of `sf.service.VideoPlayer` class, refer to [here](#).

This class describes functions for video control with service API and defines various functions that play, stop, pause, resume of video and so on.

Add the following line for `sf.service.VideoPlayer` class into a html file your own :

```
<script type="text/javascript" src="$MANAGER_WIDGET/Common/af/2.0.0/loader.js"></script>
```

You can declare `sf.service.VideoPlayer` class like this :

```
ex) var videoPlayer = sf.service.VideoPlayer;
```

## Contents

### Methods

- [focus](#)
- [getZIndex](#)
- [hide](#)
- [init](#)
- [pause](#)
- [play](#)
- [resume](#)
- [setFullScreen](#)
- [setKeyHandler](#)
- [setKeyHelp](#)
- [setPosition](#)
- [setZIndex](#)
- [show](#)
- [stop](#)

## Methods

### focus

Description

This function sets the focus to the `VideoPlayer`. This API works only in full-screen mode.

Parameters	none
Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
sf.service.VideoPlayer.focus();	

## getZIndex

### Description

This function returns z-index of VideoPlayer.

Parameters	none
Return	■Number - the z-index of the VideoPlayer.
Emulator Support	N
SDK Constraint	None
Example	
var zIndex = sf.service.VideoPlayer.getZIndex();	

## hide

### Description

This function hides the VideoPlayer.

Parameters	none
Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
sf.service.VideoPlayer.hide();	

## init

### Description

This function initialize the VideoPlayer with given option.

Parameters	<p>■option (Optional)</p> <ul style="list-style-type: none"> <li>- Object</li> <li>- Contains event handler</li> </ul> <pre>{   onstatechange: function(state) {...}, // called when VideoPlayer state changes   onend: function() {...},             // called when video play ends   onerror: function(error) {...}       // called when error occurs }</pre> <p>* state can be one of:</p> <ul style="list-style-type: none"> <li>- sf.service.VideoPlayer.STATE_PLAYING</li> <li>- sf.service.VideoPlayer.STATE_STOPPED</li> <li>- sf.service.VideoPlayer.STATE_PAUSED</li> <li>- sf.service.VideoPlayer.STATE_BUFFERING</li> <li>- sf.service.VideoPlayer.STATE_SCANNIG</li> </ul>
------------	---

Return	■Void
Emulator Support	N
SDK Constraint	None

#### Example

```
sf.service.VideoPlayer.init({
  onstatechange: function (state) {
    alert('Current State : ' + state);
  },
  onend: function(){
    alert('Video ended.');
```

```
  },
  onerror: function (error) {
    alert('Error : ' + error);
  }
});
```

## pause

#### Description

This function pauses the video.

Parameters	none
Return	■Boolean - true : Succeeded - false : fail
Emulator Support	N
SDK Constraint	None

#### Example

```
sf.service.VideoPlayer.pause();
```

## play

#### Description

This function plays the video with the specified given option.

Parameters	■option - Object - Contains the video information { url: 'http://www.samsung.com/samsung.mkv', Video URL. title: 'Samsung movie' , Video title. } This will be shown on full screen information bar.
Return	■Void
Emulator Support	N
SDK Constraint	None

#### Example

```
sf.service.VideoPlayer.play(item);
```

## resume

#### Description

This function resumes the video.

Parameters	none
------------	------

Return	<b>■</b> Boolean - true : Succeeded - false : fail
Emulator Support	N
SDK Constraint	None
Example	
<pre>sf.service.VideoPlayer.resume();</pre>	

## setFullScreen

### Description

This function sets or unsets the full-screen mode. In the full-screen mode, VideoPlayer takes key handling from scene. Add key handlers using setKeyHandler().

Parameters	<b>■</b> bOn - Boolean - true : set full-screen, false : unset full-screen
Return	<b>■</b> Void
Emulator Support	N
SDK Constraint	None
Example	
<pre>sf.service.VideoPlayer.setFullScreen(true); sf.service.VideoPlayer.setFullScreen(false);</pre>	

## setKeyHandler

### Description

This function sets the key handler. This function is for full-screen mode only.

Parameters	<b>■</b> key - Number - sf.key value <b>■</b> handler - Function - The key handler function. If the given key is pressed, this function is called.
Return	<b>■</b> Void
Emulator Support	N
SDK Constraint	None
Example	
<pre>sf.service.VideoPlayer.setKeyHandler(sf.key.RETURN, function () {     sf.service.VideoPlayer.stop(); });</pre>	

## setKeyHelp

### Description

This function sets key help description to KeyHelp Bar on full-screen mode.

Parameters	<b>■</b> key - Number - sf.key value. <b>■</b> desc - String - key description.
------------	--

Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
sf.service.VideoPlayer.setKeyHelp(sf.key.TOOLS, 'Tools');	

## setPosition

### Description

This function sets the VideoPlayer position at the specified rect. The VideoPlayer information panel is shown inside the rect. It takes some of the rect base.

Parameters	<b>■rect</b> - Object - The position information <pre>{   left: 100, left-coordinate of VideoPlayer area.   top: 100, top-coordinate of VideoPlayer area.   width: 500, width of VideoPlayer area.   height: 400, height of VideoPlayer area. }</pre> VideoPlayer takes some of the area base to show information panel (73px for 540p).
Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
sf.service.VideoPlayer.setPosition(this.videoPos);	

## setZIndex

### Description

This function sets z-index of VideoPlayer.

Parameters	<b>■zIndex</b> - Number - z-index
Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
sf.service.VideoPlayer.setZIndex();	

## show

### Description

This function shows the VideoPlayer.

Parameters	none
Return	■Void
Emulator Support	N
SDK Constraint	None

Example	
<pre>sf.service.VideoPlayer.show();</pre>	
<b>stop</b>	
Description	
This function stops playing video.	
Parameters	none
Return	■Void
Emulator Support	N
SDK Constraint	None
Example	
<pre>sf.service.VideoPlayer.stop();</pre>	