

How to implement correctly the Mute function?

Published 2014-10-28 | (Compatible with SDK 4.5,5.0,5.1 and 2013,2014 models)

Implementation of mute functions in applications with custom volume control

In case you are implementing your own custom volume control for your application, please keep in mind the following rules when developing the mute functions:

1. To get or set device mute status, please use the Media API `SetMute` and `GetMute`. If functions `SetMute` or `GetMute` are used instead, the mute status may not be saved after application exit, what is considered as app defect.
2. In order to make sure the mute functions are working on all platforms, apply only Number values (0 and 1) to the `SetMute` function. The 2010 and 2011 browser engine casts the values automatically, however the Webkit engine on 2012 platform does not work properly in this case. For that reason the native plugin Audio may not recognize these arguments properly on 2012 platforms.

For compatibility with all the platforms, it is recommended to replace all occurrences of

```
webapis.audiocontrol.setMute(true);  
...  
webapis.audiocontrol.setMute(false);
```

```
try(  
  if(webapis.audiocontrol.getMute){  
    console.log("tv is on mute");  
  }else{  
    console.log("tv is not on mute");  
  }  
}catch(error){  
  console.log(error.name);  
}
```

with

```
webapis.audiocontrol.setMute(1);  
...  
webapis.audiocontrol.setMute(0);  
  
try(  
  if(webapis.audiocontrol.getMute){  
    console.log("tv is on mute");  
  }else{  
    console.log("tv is not on mute");  
  }  
}catch(error){  
  console.log(error.name);  
}
```