

Example

# Set a playback's lock state

<b>by:</b>	Alex del Bondio/Gregory Haynes
<b>published:</b>	here, August 2018
<b>description:</b>	set a playback's lock state (unlocked/locked/transparent)
<b>remarks:</b>	beautifully explains how to set the type a property requires

[set](#), [playback](#), [lock](#), [state](#)

## functions

- [Handles.SetSourceHandle](#)
- [Handles.SetLockState](#)
- [Math.ToEnum](#)
- [Handles.ClearSelection](#)
- [ActionScript.SetProperty.Enum](#)

## affected properties

- [Handles.SourceHandle.LockState](#)

## Code

[pb1lockstates.xml](#)

```
<?xml version="1.0" encoding="utf-8"?>
<avolites.macros>

<!-- set PBI to unlocked -->
<macro id="adb.Macros.pbunlocked" name="PB unlocked">
  <sequence>
    <step>Handles.SetSourceHandle("PlaybackWindow", 0)</step>
    <step>Handles.SetLockState(
      Math.ToEnum("Avolites.Titan.Controllers",
        "Avolites.Titan.Controllers.Handles.HandleGroup+LockStates",
        "Unlocked")
    )</step>
    <step>Handles.ClearSelection()</step>
  </sequence>
</macro>

<!-- set PBI to locked -->
<macro id="adb.Macros.pblocked" name="PB locked">
  <sequence>
    <step>Handles.SetSourceHandle("PlaybackWindow", 0)</step>
```

```
<step>Handles.SetLockState(  
    Math.ToEnum("Avolites.Titan.Controllers",  
"Avolites.Titan.Controllers.Handles.HandleGroup+LockStates",  
"FullyLocked")  
    )</step>  
    <step>Handles.ClearSelection()</step>  
</sequence>  
</macro>  
  
<!-- set PB1 to transparent lock -->  
<macro id="adb.Macros.pbtranslocked" name="adb transparent lock">  
    <sequence>  
        <step>Handles.SetSourceHandle("PlaybackWindow", 0)</step>  
        <step>Handles.SetLockState(  
            Math.ToEnum("Avolites.Titan.Controllers",  
"Avolites.Titan.Controllers.Handles.HandleGroup+LockStates",  
"TransparentLocked")  
        )</step>  
        <step>Handles.ClearSelection()</step>  
    </sequence>  
</macro>  
  
<!-- alternative syntax for setting locked -->  
<macro id="adb.Macros.pblocked1" name="adb locked 1">  
    <sequence>  
        <step>Handles.SetSourceHandle("PlaybackWindow", 0)</step>  
<step>ActionScript.SetProperty.Enum("Handles.SourceHandle.LockState",  
"FullyLocked")</step>  
        <step>Handles.SetLockState(Handles.SourceHandle.LockState)</step>  
        <step>Handles.ClearSelection()</step>  
    </sequence>  
</macro>  
</avolites.macros>
```

## Explanation

This explains the functional steps within the sequence. For all the other XML details please refer to [Formats and syntax](#)

- `Handles.SetSourceHandle("PlaybackWindow", 0)` selects a certain handle (playback window button #1) for future operations
- `Handles.SetLockState(. .)` sets this playback's lock state
- `Handles.ClearSelection()` deselects the playback handle

The more interesting part is how the actual lock state is defined. Internally, the lockstate can assume one of three values: "Unlocked", "FullyLocked", and "TransparentLocked". However, the function

Handles.SetLockState() expects the state as [Enum](#).

In the first three macros this enum is temporarily created and passed to the function:

```
Handles.SetLockState(  
    Math.ToEnum(  
        "Avolites.Titan.Controllers",  
        "Avolites.Titan.Controllers.Handles.HandleGroup+LockStates",  
        "Unlocked"  
    )  
)
```

- `Avolites.Titan.Controllers` is the assembly - let's regard this as the context in which this is supposed to live
- `Avolites.Titan.Controllers.Handles.HandleGroup+LockStates` is the full name of the type
- `Unlocked` is the value we want to assign here

In the fourth macro, instead of this temporary solution a property is used (and this is the point: this way is only available if such a property exists):

- `ActionScript.SetProperty.Enum("Handles.SourceHandle.LockState", "FullyLocked")` sets a property to the enum representation of "FullyLocked"
- `Handles.SetLockState(Handles.SourceHandle.LockState)` uses this enum as parameter for `SetLockState()`

Setting the property on its own is not enough as it is only used for display purposes and does not update the handle properties when it is changed. Since the `SetLockState` function requires an Enum type and not a string you need to either create an Enum value using `Math.ToEnum` or set `Handles.SourceHandle.LockState` to the correct value and pass that in.

## How to use it

- [make this macro available](#)
- this might be useful in the context of macros to prepare customized workspaces, e.g. [Create Workspaces](#)

From:  
<https://avosupport.de/wiki/> - **AVOSUPPORT**

Permanent link:  
<https://avosupport.de/wiki/macros/example/setplaybacklockstate?rev=1545584716>

Last update: **2018/12/23 17:05**

