Example

Programmer - Direction

by:	Alex del Bondio
published:	January 2019
description:	Changes programmer direction
remarks:	

adb_programmerdirection.xml

programmer, direction, overlap

This selection of macros lets you change the programmer direction to be used with fixture overlap. These macros have Active Binding set so they will highlight when the corresponding attribute is set.

The code below is a snippet of the whole file which contains macros for all available directions.

functions

- ActionScript.SetProperty.Enum
- Active Binding

affected properties

- Selection.Context.Global.Direction

Code

direction.xml

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


<!-->Directions: None, LeftToRight, TopDown, HorizontalBlinds,
RightToLeft, BottomUp, VerticalBlinds, TopLeftIn, TopRightIn,
RadialMiddle, BottomLeftIn, BottomRightIn, BottomMiddleUpRadial,
EndsInLinear, Random<-->

    <macro id="adb.Macros.direct.LeftToRight" name="Direction
LeftToRight">
            <active binding="{propertyLink
id='Selection.Context.Global.Direction'
converter='Math.EnumAsStringEqualityConverter'
converterParameter='LeftToRight'}"/>
```

```
                <sequence>
<step>ActionScript.SetProperty.Enum("Selection.Context.Global.Direction
", "LeftToRight")</step>
        </sequence>
    </macro>


    <macro id="adb.Macros.direct.RightToLeft" name="Direction
RightToLeft">
            <active binding="{propertyLink
id='Selection.Context.Global.Direction'
converter='Math.EnumAsStringEqualityConverter'
converterParameter='RightToLeft'}"/>
            <sequence>
<step>ActionScript.SetProperty.Enum("Selection.Context.Global.Direction
", "RightToLeft")</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

- `Selection.Context.Global.Direction` is the property regarding the direction and needs to be set via `ActionScript.SetProperty.Enum`
- `active binding` is used to highlight a macro when the corresponding property (`propertyLink id`) matches the `converterParameter`. A good starting point to understand active binding is either this wiki or this post in the Avo forum http://forum.avolites.com/viewtopic.php?f=20&t=5962

How to use it

- make this macro available

From:
https://www.avosupport.de/wiki/ - **AVOSUPPORT**

Permanent link:
https://www.avosupport.de/wiki/macros/example/programmerdirection?rev=1547800899

Last update: **2019/01/18 08:41**