

Example

# Store cues with current page names as legends

<b>by:</b>	Sebastian Beutel
<b>published:</b>	May 2024
<b>description:</b>	sore cues with current pagenames as legends
<b>remarks:</b>	<p>This records some cues into the Macros window and sets their legends to the current pages legends of some windows. Page legends are not available through WebAPI. In order to get these legends:</p> <ul style="list-style-type: none"><li>- deploy this macro</li><li>- recall it from WebAPI</li></ul> <p><a href="http://localhost:4431/titan/script/2/UserMacros/RecallMacroById?macroId=Wiki.Macros.Test.PageName">http://localhost:4431/titan/script/2/UserMacros/RecallMacroById?macroId=Wiki.Macros.Test.PageName</a></p> <ul style="list-style-type: none"><li>- get handles JSON <a href="http://localhost:4431/titan/handles/Macros">http://localhost:4431/titan/handles/Macros</a></li><li>- parse the answer for the numbers/legends you are after</li></ul>

[pagename](#), [webapi](#), [trick](#), [json](#)

## functions

- [ActionScript.SetProperty](#)
- [Handles.Macros.ChangePage](#)
- [Handles.SetSourceHandleFromHandle](#)
- [Handles.ConfirmDelete](#)
- [Math.ToString](#)
- [Playbacks.StoreCue](#)
- [Handles.SetSourceHandle](#)
- [Handles.SetUserNumber](#)
- [Handles.ClearSelection](#)

## affected properties

- [UserMacros.CurrentUserNumber](#)
- [Handles.Macros.Page.Index](#)
- [Playbacks.PendingLegend](#)
- [Handles.Playbacks.Page.DisplayName](#)
- [Handles.CurrentUserNumber](#)
- [Handles.Playbacks.Page.DisplayIndex](#)

File with some more windows' pages here:

[pagename.xml](#)

## Code

[filename.xml](#)

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

```
<avolites.macros>

  <macro id="Wiki.Macros.Test.PageName" name="PageName">
    <sequence>

      <step>ActionScript.SetProperty("UserMacros.CurrentUserNumber",
Handles.Macros.Page.Index)</step>
      <step>Handles.Macros.ChangePage(1)</step>

<step>Handles.SetSourceHandleFromHandle("cueHandleUN=1001")</step>
      <step>Handles.ConfirmDelete()</step>
      <step>ActionScript.SetProperty("Playbacks.PendingLegend",
Handles.Playbacks.Page.DisplayName)</step>
      <step>Playbacks.StoreCue("Macros", 0, false)</step>
      <step>Handles.SetSourceHandle("Macros", 0)</step>
      <step>ActionScript.SetProperty("Handles.CurrentUserNumber",
userNumber:1001)</step>
      <step>Handles.SetUserNumber()</step>

<step>Handles.SetSourceHandleFromHandle("cueHandleUN=1002")</step>
      <step>Handles.ConfirmDelete()</step>
      <step>ActionScript.SetProperty("Playbacks.PendingLegend",
Math.ToString(Handles.Playbacks.Page.DisplayIndex))</step>
      <step>Playbacks.StoreCue("Macros", 1, false)</step>
      <step>Handles.SetSourceHandle("Macros", 1)</step>
      <step>ActionScript.SetProperty("Handles.CurrentUserNumber",
userNumber:1002)</step>
      <step>Handles.SetUserNumber()</step>

      <step>Handles.ClearSelection()</step>
<step>Handles.Macros.ChangePage(UserMacros.CurrentUserNumber)</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](#)

- the first two lines cache the current page of the Macros window into a variable and toggle to another page. This is to make sure all created dummy playbacks are not in the way for other things.
- then, per item (window pagename and window page number), a designated playback is stored with that value as legend, like this:

- `Handles.SetSourceHandleFromHandle(..)` selects the handle for the following delete action
  - `Handles.Delete()` deletes this handle
  - `ActionScript.SetProperty("Playbacks.PendingLegend", Handles.Playbacks.Page.DisplayName)` loads the current pagename of a window into the variable `Playbacks.PendingLegend`
  - `Playbacks.StoreCue("Macros", 0, false)` stores a dummy cue on the current Macros page's first button and sets its legend
  - `Handles.SetSourceHandle("Macros", 0)` selects this handle to set its usernumber
  - `ActionScript.SetProperty("Handles.CurrentUserNumber", userNumber:1001)` prepares the usernumber to be set for the specific handle
  - `Handles.SetUserNumber()` sets the usernumber
- the last two lines toggle the Macros window back to the page it was previously on

## How to use it

1. [make this macro available](#)

tbd

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

Permanent link:  
<https://www.avosupport.de/wiki/macros/example/pagenamestolegends?rev=1723212292>

Last update: **2024/08/09 14:04**

