

register

Ralph Wermke

COLLABORATORS

	<i>TITLE :</i> register		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Ralph Wermke	January 2, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	register	1
1.1	Welcome	1
1.2	Constuctors/ Destructor	1
1.3	New Methods	2
1.4	Tags	3
1.5	Exeptions	4
1.6	History	4
1.7	ToDo	4
1.8	Thanks	4
1.9	Author	5

Chapter 1

register

1.1 Welcome

```
register PLUGIN
```

```
Version: 1.0 beta  
Date   : 03-Sep-1997
```

Makes a set of registers/tabs for page selection.
(most useful with `changeui()`)

Constructors/ Destructor

New Methods

Special Tags

Exceptions

History

ToDo

Thanks

Author

1.2 Constructors/ Destructor

Constructor

```
register(  
    tags  
    :PTR TO tagitem)
```

For creating a new plugin object use for example:

```
DEF r:PTR TO register
NEW r.register([..., TAG_DONE])
```

1.3 New Methods

```
    set(
      tag
      , value)
```

By calling this method it's possible to change attributes at runtime. You can use all tags with the S flag set. This method can also be used before the GUI is created and when the window is closed. Value is a LONG and contains the argument for the used tag.

Example:

```
DEF r:PTR TO register
...
NEW r.register([PLA_Register_Titles, ['One','Two'], TAG_DONE])
...
r.set(PLA_Register_Disabled, TRUE)
...
```

```
value,check:=get(
  tag
)
```

This method is the counterpart to set. All tags with G flag can be used. Argument is the tag you want to get. Return values are the requested value and as second a boolean value. So if check is FALSE the used tag can't be get.

```
...
value,check:=r.get(PLA_Register_Disabled)
...
```

After this:

```
value=TRUE
check=TRUE
```

But if you try:

```
...
value,check:=r.get(PLA_Register_Titles)
...
```

Then you get this:

```
value=-1
check=FALSE
```

```
draw()          /* Private */

drawRegister() /* Private */
```

1.4 Tags

I = The letters [ISG] show you when the tags can be used.

```

Initialisation
  S =
Set Method
  G =
Get Method
  PLA_Register_Disabled [ISG]
```

Disable or enable a gadget. Setting this tag causes the gadget to become disabled, it gets a ghost pattern and doesn't respond to user input any longer.

PLA_Register_ActivePage [ISG]

Set (or get) the active page of the register.
The value may range from 0 to number of titles-1.

Or if you're using the set method it's also possible to use these special values:

```

PLV_Register_ActivePage_First
PLV_Register_ActivePage_Last
PLV_Register_ActivePage_Next
PLV_Register_ActivePage_Prev
```

PLA_Register_Titles [I..]

List of strings describing the titles of your register.
This tag MUST be set! Empty lists are also NOT allowed and will raise the

```

exception TITL
. Empty strings were handled without
problems.
```

Example:

```
PLA_Register_Titles, ['Page1', 'Page2', 'Page3'],
```

PLA_Register_ActionHook [I..]

If you want to react on all changes, made by user input AND the set method, you can use this tag.

Argument for this tag is a pointer to a installed hook.

This tag will be called whenever the active page will be changed.

The hook function will be called with 3 arguments:

- pointer to your hook function
- pointer to the register object
- number of the active page.

Example:

```
PROC actionhook(hook:PTR TO hook, obj:PTR TO register, activepage)
...
ENDPROC
```

1.5 Exeptions

Constructor

"UTIL" will be raised if the utility.library v37+ can't be opened.
"TITL" will be raised if the tiles are not set correctly

1.6 History

03-Sep-97
- first beta version

1.7 ToDo

- setable colors
- add lists of GUIs to register
- better update (less flicker)

1.8 Thanks

Special thanks go to:

Stefan Blixth for beta testing

1.9 Author

Contact me for bugreports, suggestions or whatever you want:

SnailMail:

Ralph Wermke
Prokofjewstrasse 7
17491 Greifswald
Germany

EMail:

wermke@gryps1.rz.uni-greifswald.de

WWW:

<http://www.user.fh-stralsund.de/~rwormke/di.html>
