

corners

COLLABORATORS

| | | | |
|------------|--------------------|-----------------|-----------|
| | TITLE : corners | | |
| ACTION | NAME | DATE | SIGNATURE |
| WRITTEN BY | | August 22, 2024 | |

REVISION HISTORY

| | | | |
|--------|------|-------------|------|
| NUMBER | DATE | DESCRIPTION | NAME |
| | | | |

Contents

| | | |
|----------|---|----------|
| 1 | corners | 1 |
| 1.1 | corners_plugin: Introduction | 1 |
| 1.2 | corners_plugin: Constructors / Destructor | 1 |
| 1.3 | corners_plugin: New Methods | 2 |
| 1.4 | corners_plugin: Tags | 2 |
| 1.5 | corners_plugin: Exceptions | 3 |
| 1.6 | corners_plugin: History | 4 |

Chapter 1

corners

1.1 corners_plugin: Introduction

corners_plugin

by Ali Graham <agraham@hal9000.net.au>

corners_plugin is a PLUGIN which enables you to choose between the four corners of a square (envisaged usage; screen blanker preferences program). While maybe not immediately useful to the general public, it illustrates the usage of detecting input in different areas of a PLUGIN.

Constructor

Methods

Tags

Exceptions

History

1.2 corners_plugin: Constructors / Destructor

Constructor

```
corners(tags=NIL:PTR TO tagitem)
```

For creating a new plugin object use for example:

```
DEF corners:PTR TO corners_plugin  
NEW corners.corners([..., TAG_DONE])
```

1.3 corners_plugin: 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 corners:PTR TO corners_plugin
...
NEW corners.corners([..., TAG_DONE])
...
corners.set(PLA_Corners_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:=corners.get(PLA_Corners_Disabled)
...
```

After this:

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

But if you try:

```
...
value,check:=corners.get(PLA_Corners_Width)
...
```

Then you get this:

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

```
draw()          /* Private */
```

1.4 corners_plugin: Tags

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

I = Initialisation

S = Set Method

G = Get Method

```
PLA_Corners_Width          [I..]
& PLA_Corners_Height       [I..]
```

The size of the overall PLUGIN. Remember that each of these should be over twice the size of the corresponding BoxWidth/BoxHeight tag (no checks are performed in the PLUGIN for this).

```
PLA_Corners_BoxWidth       [ISG]
& PLA_Corners_BoxHeight    [ISG]
```

The size of the smaller box, that denotes which corner is active.

```
PLA_Corners_ActiveCorner   [ISG]
```

This can be one of five values:

```
PLV_Corners_CornerNone
PLV_Corners_CornerTopLeft
PLV_Corners_CornerTopRight
PLV_Corners_CornerLowerLeft
PLV_Corners_CornerLowerRight
```

```
PLA_Corners_FGColor        [ISG]
& PLA_Corners_BGColor      [ISG]
```

The pen colours of the 'active-corner' box (FGColor) and the background of the PLUGIN (BGColor). (I'd normally spell that with a "u", but I'm following the spelling in the Includes, here.)

```
PLA_Corners_Disabled       [ISG]
```

Disable or enable the PLUGIN. Setting this tag causes the PLUGIN to become disabled; it gets a ghost pattern.

1.5 corners_plugin: Exceptions

Constructor

"util" will be raised if the utility.library has not been opened.

1.6 corners_plugin: History

v1.0 (18.2.98)

- o Initial release.

v1.1 (25.2.98)

- o Replaced the CORNER_#? CONST's with proper EPSG PLV_Corners_#? values.
-