

MiscClipTextField

Inherits From: TextField : Control : View : Responder : Object
Declared In: MiscClipTextField.h
Depends On: MiscClipTextFieldCell.h

Class Description

MiscClipTextField is a TextField object which can display long string values that normally would be truncated because of short of space. If a given string value is too long to be displayed in whole MiscClipTextField clips the string at a reasonable point and puts a user defined string (clipper) in the place of the clipped string part.

MiscClipTextField heavily depends on MiscClipTextFieldCell and most of its methods are provided for easier use only, they do nothing only pass the message further to cell. There are two things MiscClipTextField actually does: handles triple-click mouse events and updates its cell's string value when resized.

When a triple-click mouse event occurs in a MiscClipTextFieldCell the cell of the field becomes

selectable and scrollable and you can cut/copy the original (full) string value.

When the MiscClipTextFieldCell is resized it simply messages its cell to reset its string value (ie: clip it again according to the new size).

Known Bugs

If a MiscClipTextField is set to display its text right aligned or centered and the user triple-clicks on it the field changes to display its text left aligned. Then when you try to resize the view the first time, it doesn't change back to the original alignment (I don't really know why), but it does at the second time (???)

Anyway it doesn't seem to have dangerous bugs in it, and should work as advertised :-)

Instance Variables

None declared in this class.

Method Types

- clipper
- delimiters
- doesClipOnRight
- fullStringValue

- getInspectorClassName
- initWithFrame:
- isClipEnabled
- mouseDown:
- resetStringValue:
- + setCellClass:
- setClipDelimiters:
- setClipEnabled:
- setClipOnRight:
- setClipperString:
- sizeTo::

Class Methods

setCellClass:

+ **setCellClass:***classId*

Method description here.

See also: - **myReference**

Instance Methods

changeFont:

- **changeFont:***sender*

Method description here.

See also: - **myReference**

clipper

- **clipper**

Method description here.

See also: - **myReference**

delimiters

- **delimiters**

Method description here.

See also: - **myReference**

doesClipOnRight

- (BOOL) **doesClipOnRight**

Method description here.

See also: - **myReference**

fullStringValue

- (const char*)**fullStringValue**

Method description here.

See also: - **myReference**

getInspectorClassName

- (const char *)**getInspectorClassName**

Method description here.

See also: - **myReference**

initWithFrame:

- **initWithFrame:**(const NXRect *)*rect*

Method description here.

See also: - **myReference**

isClipEnabled

- (BOOL) **isClipEnabled**

Method description here.

See also: - myReference

mouseDown:

- **mouseDown:**(NXEvent *)*event*

Method description here.

See also: - myReference

resetStringValue:

- **resetStringValue:***sender*

Method description here.

See also: - myReference

setClipDelimiters:

- **setClipDelimiters:**(const char*)*delimiters*

Method description here.

See also: - myReference

setClipEnabled:

- **setClipEnabled:**(BOOL)*flag*

Method description here.

See also: - **myReference**

setClipOnRight:

- **setClipOnRight:**(BOOL)*flag*

Method description here.

See also: - **myReference**

setClipperString:

- **setClipperString:**(const char*)*aString*

Method description here.

See also: - **myReference**

sizeTo::

- **sizeTo:**(NXCoord)*width* :(NXCoord)*height*

Method description here.

See also: - **myReference**