T

Description

This function provides a simple test of the multiautogui.

1.4.21 multiscale - Tool

Package display
Module widgets

Glish/Tk widget to specify multiple values in a single domain
include "multiscale.g"

Constructors

multiscale	*Construct a multiscale*

Functions

disable	*Disable widget*
dismiss	*Disable and dismiss the multiscale*
enable	*Enable widget*
getvalues	*Retrieve the current scale value/s*
setrange	*Change the lower and upper limits of the scales*
setresolution	*Set the step value for the scales*
setvalues	*Set the scale value/s*

Description

Note: the following information describes what the multiscale widget *should* do. At present, only one scale is supported.
This Glish/Tk widget provides one or more basic Glish/Tk scale widgets having identical value ranges, but different values. Its intelligence lies in its ability to ensure that values remain ordered. That is, it can be used for the selection of minimum and maximum values of data, and ensures that the minimum is always less than or equal to the maximum.
Whenever the user modifies any one of the scales in the multiscale widget, an event "values" is emitted from the multiscale, whose value is a vector of floats being the position of each slider.
If you chooser to show an entry box as well (`entry=T`*) as the slider, then they are*

connected. The values selected with the slider show in the entry box. When you enter a value in the entry box, after you press carriage return, the slider will update its value.
You can also interactively extend the range of the slider (`extend=T` and `entry=T`). In this case, a check box is offered. If checked, then any value you enter in the entry box which is outside of the current slider range, will extend the slider range.

multiscale - Constructor

Package	**display**
Module	**widgets**
Tool	**multiscale**

Construct a multiscale

Synopsis

multiscale(parent, start, end, values, names, helps, constrain, entry, extend, length, resolution, orient, width, font, relief, borderwidth, foreground, background, fill, widgetset)

Arguments

parent	*Parent of the label* *Allowed: Tk frame*
start	*Lower limit of range* *Allowed: float* *Default: 0.0*
end	*Upper limit of range* *Allowed: float* *Default: 100.0*
values	*initial values* *Allowed: vector of floats* *Default: [0.0, 100.0]*
names	*text labels for each scale* *Allowed: vector of strings* *Default: ['Minimum', 'Maximum']*
helps	*popup help text for each label* *Allowed: vector of strings* *Default: ""*
constrain	*whether to enforce ordering of numbers* *Allowed: T F* *Default: T*
entry	*provide an entry box as well* *Allowed: T F* *Default: F*
extend	*extend the scale if entered value beyond current extrema* *Allowed: T F* *Default: F*
length	*length of the scale widgets* *Allowed: dimension* *Default: 110*
resolution	*step value along scales* *Allowed: float* *Default: 1.0*
orient	*orientation of scale* *Allowed: 'horizontal' 'vertical'* *Default: 'horizontal'*
width	*transverse width of the scales* *Allowed: dimension* *Default: 15*
font	*text font* *Allowed: X font* *Default: "*
relief	*border relief* *Allowed: ‘flat’ ‘ridge’, ‘raised’ ‘sunken’ ‘groove’* *Default: 'flat'*
borderwidth	*border width* *Allowed: dimension* *Default: 2*
foreground	*color of text* *Allowed: X color* *Default: ‘black’*
background	*background color* *Allowed: X color* *Default: ‘lightgrey’*
fill	*how to expand when resized* *Allowed: 'x' 'y' 'both' 'none* *Default: 'none'*
widgetset	*widgetserver to use to make label* *Allowed: widgetserver toolXS* *Default: dws*

Returns
multiscale widget

Description
Constructs a multiscale according to the specified construction arguments, and returns a ref to the widget.

Example

```
include 'multiscale.g'
f := frame();
ms := multiscale(f, start=20.0, end=55.0, resolution=0.2);
```

---

multiscale.disable - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Disable widget

Synopsis
disable()

Returns
T

Description
This function disables the widget.

Example

```
f := frame();
ms := multiscale(f);
ms.disable();
```

---

multiscale.enable - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Enable widget

Synopsis
enable()

Returns
T

Description
This function enables the widget.

Example

```
f := frame();
ms := multiscale(f);
ms.disable();
ms.enable();
```

---

multiscale.getvalues - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Retrieve the current scale value/s

Synopsis
getvalues()

Returns
vector of floats

Description
This function is used to retrieve the value of each of the scales. A vector of floats is returned.

Example

```
- ms := multiscale(spareframe);
- print ms.getvalues();
[0 100]
```

---

multiscale.setrange - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Change the lower and upper limits of the scales

Synopsis
setrange(start, end)

Arguments

start	*lower limit for scale/s* *Allowed: float* *Default: Current value*
end	*upper limit for scale/s* *Allowed: float* *Default: Current value*

Returns
none

Description
This function can be used to change the range of the scale/s after construction of the widget.

Example

```
ms := multiscale(spareframe);
ms.setrange(-30.0, 30.0);
```

---

multiscale.setresolution - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Set the step value for the scales

Synopsis
setresolution(resolution)

Arguments

resolution	*step size for scales* *Allowed: float* *Default: 1.0*

Returns
none

Description
This function should be used to modify the step size on the scale/s after construction of the widget.

Example

```
ms := multiscale(spareframe);
ms.setresolution(0.01);
```

---

multiscale.setvalues - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Set the scale value/s

Synopsis
setvalues(values)

Arguments

values	*values for the scales* *Allowed: vector of floats*

Returns
none

Description
This function should be used to change the values shown on the scale/s after construction of the widget.

Example

```
ms := multiscale(spareframe, names="bass mid treble",
                 values=[50, 50, 50], constrain=F,
                 orient='vertical', labelside='top');
ms.setvalues([10, 40, 60]);
```

---

multiscale.dismiss - Function

Package	**display**
Module	**widgets**
Tool	**multiscale**

Disable and dismiss the multiscale

Synopsis
dismiss()

Returns
None