

[back](#) [next](#)

The tkinter Scrollbar Widget

When to use the Scrollbar Widget

This widget is used to implement scrolled listboxes, canvases, and text fields.

Patterns

The **Scrollbar** widget is almost always used in conjunction with a **Listbox**, **Canvas**, or **Text** widget. Horizontal scrollbars can also be used with the **Entry** widget.

To connect a vertical scrollbar to such a widget, you have to do two things:

1. Set the widget's **yscrollcommand** callbacks to the **set** method of the scrollbar.
2. Set the scrollbar's **command** to the **yview** method of the widget.

```
from tkinter import *

master = Tk()

scrollbar = Scrollbar(master)
scrollbar.pack(side=RIGHT, fill=Y)

listbox = Listbox(master, yscrollcommand=scrollbar.set)
for i in range(1000):
    listbox.insert(END, str(i))
listbox.pack(side=LEFT, fill=BOTH)

scrollbar.config(command=listbox.yview)

mainloop()
```

When the widget view is modified, the widget notifies the scrollbar by calling the **set** method. And when the user manipulates the scrollbar, the widget's **yview** method is called with the appropriate arguments.

Adding a horizontal scrollbar is as simple. Just use the **xscrollcommand** option instead, and the **xview** method.

For more examples, see the [tkinter Scrollbar Patterns](#) article.

FIXME: Add [AutoScrollbar](#) pattern.

Reference

Scrollbar(master=None, **options) (class) [<#>]

A scrollbar.

master

Parent widget.

***options*

Widget options. See the description of the [config](#) method for a list of available options.

activate(element) [<#>]

Activates a scrollbar element.

element

The scrollbar element to activate. Use one of “arrow1”, “slider”, or “arrow2”. If omitted, the method returns the currently active element, or an empty string if no element is active.

config(options)** [<#>]

Modifies one or more widget options. If no options are given, the method returns a dictionary containing all current option values.

****options**

Widget options.

activebackground=

Default value is system specific. (the database name is activeBackground, the class is Foreground)

activerelief=

Default value is RAISED. (activeRelief/Relief)

background=

Default value is system specific. (background/Background)

bg=

Same as **background**.

borderwidth=

Border width. Default value is 0. (borderWidth/BorderWidth)

bd=

Same as **borderwidth**.

command=

A callback used to update the associated widget. This is typically the **xview** or **yview** method of the scrolled widget.

If the user drags the scrollbar slider, the command is called as

callback("moveto", offset), where offset 0.0 means that the slider is in its topmost (or leftmost) position, and offset 1.0 means that it is in its bottommost (or rightmost) position.

If the user clicks the arrow buttons, or clicks in the trough, the command is called as **callback("scroll", step, what)**. The second argument is either "-1" or "1" depending on the direction, and the third argument is "units" to scroll lines (or other unit relevant for the scrolled widget), or "pages" to scroll full pages. (command/Command)

cursor=

The cursor to show when the mouse pointer is placed over the scrollbar widget. Default is a system specific arrow cursor. (cursor/Cursor)

elementborderwidth=

Default value is -1. (elementBorderWidth/BorderWidth)

highlightbackground=

Together with **highlightcolor** and **highlightthickness**, this option controls how to draw the highlight border. When the widget has focus, a **highlightthickness**-wide border is drawn in the **highlightcolor** color.

Otherwise, it is drawn in the **highlightbackground** color. The default colors are system specific. (highlightBackground/HighlightBackground)

highlightcolor=

See **highlightbackground**. (highlightColor/HighlightColor)

highlightthickness=

See **highlightbackground**. Default value is 0 (no highlight border). (highlightThickness/HighlightThickness)

jump=

Default value is 0. (jump/Jump)

orient=

Defines how to draw the scrollbar. Use one of **HORIZONTAL** or **VERTICAL**. Default is **VERTICAL**. (orient/Orient)

relief=

Border decoration. The default is **SUNKEN**. Other possible values are **FLAT**, **RAISED**, **GROOVE**, and **RIDGE**. This option is ignored under Windows. (relief/Relief)

repeatdelay=

Default value is 300. (repeatDelay/RepeatDelay)

repeatinterval=

Default value is 100. (repeatInterval/RepeatInterval)

takefocus=

Default is an empty string. (takeFocus/TakeFocus)

troughcolor=

Default value is system specific. (troughColor/Background)

width=

Scrollbar width, in pixels. Default value is 16. (width/Width)

delta(deltax, deltay) [<#>]

Returns a floating point number that should be added to the current slider offsets in order to move the slider the given number of pixels. This is typically used by the mouse bindings to figure out how to move the slider when the user is dragging it around.

deltax

Horizontal delta, in screen coordinates.

deltay

Vertical delta, in screen coordinates.

Returns:

Value to add to the scrollbar offset.

`fraction(x, y)` [<#>]

Returns the slider position corresponding to a given mouse coordinate.

x

Horizontal screen coordinate.

y

Vertical screen coordinate.

Returns:

The scrollbar offset corresponding to the given coordinate (0.0 through 1.0).

`get()` [<#>]

Gets the current slider position.

Returns:

A tuple containing the relative offset for the upper (leftmost) and lower (rightmost) end of the scrollbar slider. Offset 0.0 means that the slider is in its topmost (or leftmost) position, and offset 1.0 means that it is in its bottommost (or rightmost) position.

`identify(x, y)` [<#>]

Identifies the scrollbar element at the given location.

x

Horizontal screen coordinate.

y

Vertical screen coordinate.

Returns:

One of “arrow1” (top/left arrow), “trough1”, “slider”, “trough2”, “arrow2” (bottom/right), or an empty string for any other part of the scrollbar widget.

`set(lo, hi)` [<#>]

Moves the slider to a new position.

lo

The relative offset for the upper (leftmost) end of the scrollbar slider.

hi

The relative offset for the lower (rightmost) end of the the scrollbar slider.

[back](#) [next](#)