

# RUBY/TK - PROGRESSBAR WIDGET

[http://www.tutorialspoint.com/ruby/ruby\\_tk\\_progressbar.htm](http://www.tutorialspoint.com/ruby/ruby_tk_progressbar.htm)

Copyright © tutorialspoint.com

## Description:

A **ProgressBar** provides a widget, which will show a graphical representation of a value, given maximum and minimum reference values.

## Syntax:

Here is a simple syntax to create this widget:

```
Tk::ProgressBar.new(root) {  
  ....Standard Options....  
  ....Widget-specific Options....  
}
```

## Standard Options:

- borderwidth
- highlightthickness
- padx
- pady
- relief
- troughcolor

These options have been described in previous chapter.

## Widget-specific Options:

SN	Options with Description
----	--------------------------

- |   |  |
|---|--|
| 1 | <b>anchor</b> => String<br><br>This can be used to position the start point of the bar. Default is 'w'<br><i>horizontalbarstartingfromtheleft</i> . A vertical bar can be configured by using either 's' or 'n'. |
| 2 | <b>blocks</b> => Integer<br><br>This controls the number of blocks to be used to construct the progress bar. The default is to break the bar into 10 blocks.   |
| 3 | <b>colors</b> => String<br><br>Controls the colors to be used for different positions of the progress bar.   |
| 4 | <b>from</b> => Integer<br><br>This sets the lower limit of the progress bar. If the bar is set to a value below the lower limit no bar will be displayed. Defaults to 0.   |
| 5 | <b>gap</b> => Integer<br><br>This is the spacing <i>inpixels</i> between each block. Defaults to 1. Use 0 to get a continuous bar.   |

6 **length** => Integer

Specifies the desired long dimension of the ProgressBar in screen units

7 **resolution** => Integer

A real value specifying the resolution for the scale. If this value is greater than zero, then the scale's value will always be rounded to an even multiple of this value, as will tick marks and the endpoints of the scale. Defaults to 1

8 **to** => Integer

This sets the upper limit of the progress bar. If a value is specified *forexample, using the value method* that lies above this value the full progress bar will be displayed. Defaults to 100.

9 **variable** => Variable

Specifies the reference to a scalar variable to link to the ProgressBar. Whenever the value of the variable changes, the ProgressBar will update to reflect this value.

10 **value** => Integer

The can be used to set the current position of the progress bar when used in conjunction with the standard configure. It is usually recommended to use the **value** method instead.

11 **width** => Integer

Specifies the desired narrow dimension of the ProgressBar in screen units

## Manipulating Progress Bar:

You can use **value?value?** method along with ProgressBar instance to get current value of the ProgressBar. If value is given, the value of the ProgressBar is set.

## Examples:

```
require 'tk'
require 'tkextlib/bwidget'

root = TkRoot.new
root.title = "Window"

progressBar = Tk::BWidget::ProgressBar.new(root)

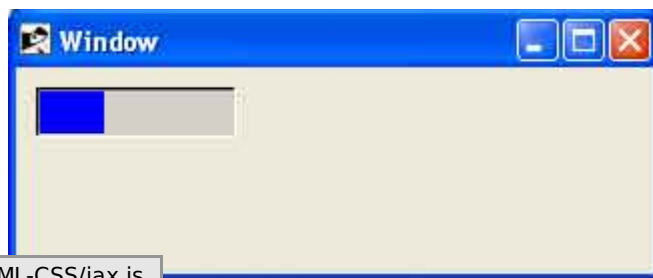
variable = TkVariable.new
progressBar.variable = variable

variable.value = 33

progressBar.maximum = 100
progressBar.place('height' => 25,
                  'width'   => 100,
                  'x'       => 10,
                  'y'       => 10)

Tk.mainloop
```

This will produce the following result:



Loading [MathJax]/jax/output/HTML-CSS/jax.js