

Description:

A *frame* is a widget that displays just as a simple rectangle. Frames are primarily used as a container for other widgets, which are under the control of a geometry manager such as grid.

The only features of a frame are its background color and an optional 3-D border to make the frame appear raised or sunken.

Syntax:

Here is a simple syntax to create a Frame Widget:

```
TkFrame.new {  
  ....Standard Options....  
  ....Widget-specific Options....  
}
```

Standard Options:

- borderwidth
- highlightbackground
- highlightthickness
- takefocus
- highlightcolor
- relief
- cursor

These options have been described in previous chapter.

Widget-specific Options:

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

1	background => String
---	-----------------------------

This option is the same as the standard **background** option except that its value may also be specified as an undefined value. In this case, the widget will display no background or border, and no colors will be consumed from its colormap for its background and border.

2	colormap => String
---	---------------------------

Specifies a colormap to use for the window. The value may be either *new*, in which case a new colormap is created for the window and its children, or the name of another window *which must be on the same screen*, in which case the new window will use the colormap from the specified window. If the **colormap** option is not specified, the new window uses the same colormap as its parent.

3	container => Boolean
---	-----------------------------

The value must be a boolean. If true, it means that this window will be used as a container in which some other application will be embedded. The window will support the appropriate window manager protocols for things like geometry requests. The window

should not have any children of its own in this application.

4 **height** => Integer

Specifies the desired height for the window in pixels or points.

5 **width** => Integer

Specifies the desired width for the window in pixels or points.

Event Bindings:

When a new frame is created, it has no default event bindings: frames are not intended to be interactive.

Examples:

```
require "tk"

f1 = TkFrame.new {
  relief 'sunken'
  borderwidth 3
  background "red"
  padx 15
  pady 20
  pack('side' => 'left')
}
f2 = TkFrame.new {
  relief 'groove'
  borderwidth 1
  background "yellow"
  padx 10
  pady 10
  pack('side' => 'right')
}

TkButton.new(f1) {
  text 'Button1'
  command {print "push button1!!\n"}
  pack('fill' => 'x')
}
TkButton.new(f1) {
  text 'Button2'
  command {print "push button2!!\n"}
  pack('fill' => 'x')
}
TkButton.new(f2) {
  text 'Quit'
  command 'exit'
  pack('fill' => 'x')
}
Tk.mainloop
```

This will produce the following result:



