RUBY/TK - PANED WINDOWS WIDGET

http://www.tutorialspoint.com/ruby/ruby_tk_paned.htm

The **Panedwindow** widget lets you stack two or more resizable widgets above and below each other *ortotheleftandright*.

The user can adjust the relative heights *orwidths* of each pane by dragging a sash located between them. Typically, the widgets you're adding to a panedwindow will be frames containing many other widgets.

Syntax:

Here is a simple syntax to create this widget:

```
Tk::Tile::Paned.new(root) {
  .....Standard Options....
  .....Widget Specific Options....
}
```

Standard Options:

- class
- cursor
- style
- takefocus

Widget-specific Options:

SN	Options with Description
1	**orient** => String One of **horizontal** or **vertical**. Specifies the orientation of the separator.
2	**width** => Integer If present and greater than zero, specifies the desired width of the widget in pixels. Otherwise, the requested width is determined by the width of the managed windows.
3	**height** => Integer If present and greater than zero, specifies the desired height of the widget in pixels. Otherwise, the requested height is determined by the height of the managed windows.

Manipulating Paned

- Calling the **"add"** method will add a new pane at the end of the list of panes. The **"insert *position subwindow*"** method allows you to place the pane at the given position in the list of panes $0..n-1$; if the pane is already managed by the panedwindow, it will be moved to the new position. You can use the **"forget *subwindow*"** to remove a pane from the panedwindow; you can also pass a position instead of a subwindow.
- Other options let you sign relative weights to each pane so that if the overall panedwindow resizes, certain panes will get more space than others. As well, you can adjust the position of each sash between items in the panedwindow. See the command reference for details.

Examples:

```
require 'tk'
require 'tkextlib/tile'

$resultsVar = TkVariable.new
root = TkRoot.new
root.title = "Window"

p = Tk::Tile::Paned.new(root)do
  height 110
  place('height' => 100, 'width' => 200, 'x' => 10, 'y' => 10)
end

f1 = TkFrame.new(p) {
  relief 'groove'
  borderwidth 3
  background "red"
  padx 30
  pady 30
  pack('side' => 'left', 'pady' => 100)
}
f2 = TkFrame.new (p){
  relief 'groove'
  borderwidth 3
  background "yellow"
  padx 30
  pady 30
  pack('side' => 'right', 'pady' => 100)
}

p.add f1, nil
p.add f2, nil

Tk.mainloop
```

This will produce the following result: