

CS 580 Client-Server Programming  
Spring Semester, 2006  
Doc 24 Some Ruby GUI 2  
Apr 27, 2006

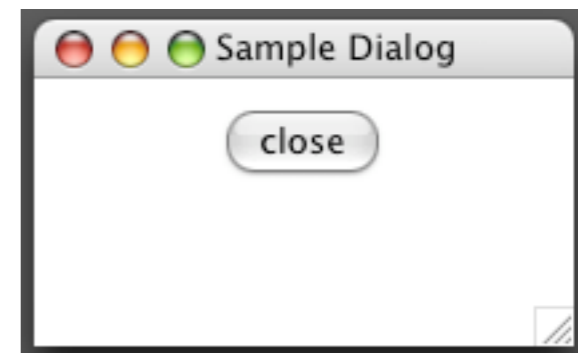
Copyright ©, All rights reserved. 2006 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

## Creating a Dialog

```
require 'tk'  
class Dialog
```



```
  def initialize  
    @@padding = { 'padx' => 10, 'pady' => 10 }  
    doFoo = proc {showDialog}  
    root = TkRoot.new { title "Simple" }  
    root.minsize(100,100)  
    root.geometry("150x100")  
    TkButton.new(root) {text 'Do It'  
      command doFoo  
      pack(@@padding)}  
    TkButton.new(root) {  
      text 'Exit'  
      command {root.destroy}  
      pack @@padding}  
  end
```



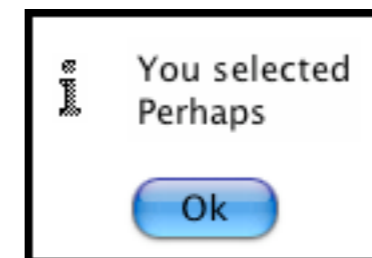
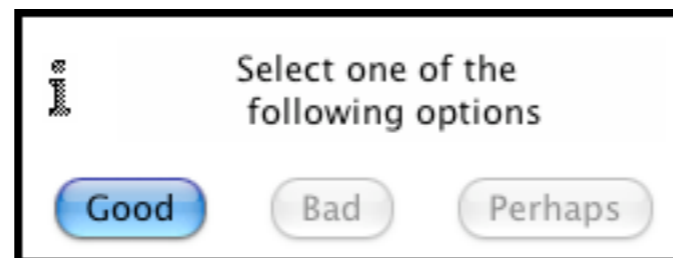
```
  def showDialog  
    popup = TkToplevel.new {|w|  
      title("Sample Dialog")  
      geometry("200x100+25+300")  
    }  
    hide = proc {popup.withdraw}  
    TkButton.new(popup) {  
      text 'close'  
      command hide  
      pack( @@padding ) }  
  end  
end  
Dialog.new  
Tk.mainloop
```

## Built-in Dialog

```
require 'tk'
class BuiltinDialog

  def foo
    options = ['Good', 'Bad', 'Perhaps']
    popup = TkDialog.new( {
      'title' => "Surprise",
      'message' => "Select one of the\n following options",
      'buttons' => options
    })
    selected = popup.value
    TkDialog.new( {
      'title' => "Result",
      'message' => ("You selected\n" + options[selected]),
      'buttons' => ['Ok']
    })
  end
end
```

```
def initialize
  pading = { 'padx' => 10, 'pady' => 10 }
  doFoo = proc {foo}
  root = TkRoot.new { title "Simple" }
  TkButton.new(root) {text 'Show It'
    command doFoo; pack pading}
  TkButton.new(root) {
    text 'Exit'
    command {root.destroy}
    pack pading}
end
end
BuiltinDialog.new
Tk.mainloop
```



## Some Ruby-Tk Examples

The examples require some tweaking

<http://www.havenrock.com/pub/ruby/tk-demos/rtkdemos-en-20020829.tar.gz>

<http://www.havenrock.com/pub/ruby/tk-demos/rtkdemos-en-20020829.zip>

# Java AWT & Threads

AWT is thread safe

# Java Swing is not Thread Safe



SwingUtilities.invokeLater(Runnable doRun)  
SwingUtilities.invokeAndWait(Runnable doRun)

## Java Swing & threads - Some references

Threads and Swing

<http://java.sun.com/products/jfc/tsc/articles/threads/threads1.html>

Using a Swing Worker Thread

<http://java.sun.com/products/jfc/tsc/articles/threads/threads2.html>

The Last Word in Swing Threads

<http://java.sun.com/products/jfc/tsc/articles/threads/threads3.html>

Java TCP Sockets and Swing Tutorial

<http://www.cise.ufl.edu/~amyles/tcpchat/>