Package ‘gWidgets2tcltk’

September 25, 2018

Version 1.0-6
Title Toolkit Implementation of gWidgets2 for tcltk
Author John Verzani
Maintainer John Verzani <jverzani@gmail.com>
URL https://github.com/jverzani/gwidgets2tcltk
Depends R (>= 2.12.0), methods, digest, memoise, tcltk(>= 2.7.0), gWidgets2(>= 1.0.7)
Suggests testthat
Description Port of the ‘gWidgets2’ API for the ‘tcltk’ package.
License GPL (>= 2)
LazyLoad yes
Collate 'List.R' 'gWidgets2tcltk-package.R' 'misc.R' 'tcltk-misc.R'
   'dnd.R' 'GComponent.R' 'GContainer.R' 'GWidget.R' 'dialogs.R'
   'gmenu.R' 'gaction.R' 'gbutton.R' 'gcalendar.R' 'gcheckbox.R'
   'gcheckboxgroup.R' 'gcombobox.R' 'gcomponent.R'
   'gdf.R' 'gdraggable.R' 'gexpanding.R' 'gfile.R' 'gformdisplay.R'
   'gframe.R' 'ggroup.R' 'gimage.R' 'glabel.R' 'glayout.R'
   'gnotebook.R' 'gpanedgroup.R' 'gprogressbar.R' 'gradio.R'
   'gseparator.R' 'gslider.R' 'gspinbutton.R' 'gstackwidget.R'
   'gstatusbar.R' 'gtextraja.R' 'gtimer.R' 'gtoplevel.R' 'gtree.R'
   'gvarbrowser.R' 'gwindow.R' 'icons.R' 'tk2tip.R' 'zzz.R'
RoxygenNote 6.1.0
NeedsCompilation no
Repository CRAN
Date/Publication 2018-09-25 13:20:02 UTC

R topics documented:

   gWidgets2tcltk-package ................................................. 2
   .gmessage.guiWidgetsToolkittcltk .................................... 4
   GComponentObservable ................................................ 12
Description

The gWidgets2 package provides a programming interface for making graphical user interfaces within R. The package is a rewrite of the gWidgets package, introducing a few external changes but a significant number of internal ones. The package relies on one of several underlying toolkit packages providing access to the graphical libraries. This package provides the implementation to interface with the underlying tcltk package.

Drag and drop in tcltk is not supported by any underlying toolkit functionality. As such, in gWidgets2 we roll our own. The result is a little limited: a) you can’t drop values from other applications b) drop targets aren’t allowed to be picky about what they receive (not by mime-type anyways).

GComponent is a parent class for both GContainer and GWidget and inherits its primary interface from gWidgets2::BasicToolkitInterface.

The gdf class provides a means to edit a data frame. We use the add on TK code provided by tablelist as the underlying widget

The main reference methods GTimer are start_timer and stop_timer

Arguments

... passed to constructor

Details

To implement drag and drop, we bind to the toplevel window the events: button1, motion and button1 release. The binding occurs not to to widget, but to the toplevel window containing the widget. This has some advantages, most importantly it is unlikely the binding will be overwritten by usual gWidgets2tcltk programs. (Recall tkbind will only allow one binding per widget per signal).

The gWidgets2 way of doing drag and drop is implemented here. You add a drop source with a handler that returns the value of what you want to pass via the dnd process. Then you specify a
widget as a drop target and give a handler. This handler receives the data through the dropdata component of the "h" argument. Here, the value is generated when the drop occurs, not when the drag initiates. Not sure this makes any difference, but it might.

Simply click on a row and the editor pops up as a modal dialog. The shortcut Shift+Enter will go onto the next case, saving changes provided the auto save feature is employed.

There is no undo/redo support here. There is no support for editing rownames (just lazy at the moment, holler if you would like that). No support to change the dimensions of the data frame or edit factors, ...

Methods

is_watching() Are we watching for a drag?
add_drag_motion(handler, action = NULL, ...) Called when motion over widget occurs
add_to_parent(parent, child, expand = NULL, fill = NULL, anchor = NULL, ...) Add a child to parent if it is a container and non null. Dispatches to add_child method of parent
get_block() Return surround block
get_enabled() is widget sensitive to user input
get_index(drop = NULL, ...) svalue; index=TRUE
get_tk_id() Return tk ID
get_toplevel_tk_id() return id of toplevel
get_value(drop = NULL, ...) Get main value of widget. From 'svalue' when index = FALSE or NULL
get_widget() Return widget (not block)
is_tkwidget() Is widget older style widget
is_ttkwidget() Is widget new style widget?. Override in subclass if not
set_enabled(value, ...) specify with logical if widget is sensitive to user input
set_size(value, ...) Set widget size (size request), value=c(width=-1, height=-1)
set_value(value, ...., drop = NULL) for 'svalue<- when index = FALSE or NULL
add(...) add is just add_child
child_bookkeeping(child) Update parent property of child and children property of parent container
get_widget() Return widget (not block)
connect_to_toolkit_signal(signal, decorator, emitter = handler_widget(), ...) Connect signal of toolkit to notify observer
remove_border() Remove border by setting relief to none
get_length(...) Get length of object. Needed for sapply.
save_data(nm, where) Save data set
set_interval(ms) Set the interval. Need to stop and start active timer to implement.
start_timer() Start the timer
stop_timer() stop the timer
Author(s)

John Verzani <jverzani@gmail.com>
Maintainer: John Verzani <jverzani@gmail.com>

Description

toolkit implementation for gmessage
toolkit implementation for gconfirm
toolkit implementation of ginput
toolkit implementation
toolkit implementation of galert
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit XXX constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit gedit constructor
toolkit constructor
Toolkit implementation
Toolkit constructor
Toolkit constructor
toolkit constructor for ggroup
gframe constructor
Toolkit constructor
Toolkit label constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit constructor
Toolkit XX constructor
Toolkit constructor
Toolkit constructor

Toolkit XXX constructor

Toolkit constructor

toolkit implementation of gtext
S3 method for gtimer
Toolkit constructor
Toolkit constructor

Toolkit constructor for gwindow
add stock icons

Returns list of stock ids
return stock id
return stock id from object

Usage

```r
## S3 method for class 'guiWidgetsToolkittcltk'
.gmessage(toolkit, msg,
    title = "message", icon = c("info", "warning", "error", "question"),
    parent = NULL, ...)

## S3 method for class 'guiWidgetsToolkittcltk'
.gconfirm(toolkit, msg,
    title = "Confirm", icon = c("info", "warning", "error", "question"),
    parent = NULL, ...)

## S3 method for class 'guiWidgetsToolkittcltk'
.ginput(toolkit, msg, text = "",
    title = "Input", icon = c("info", "warning", "error", "question"),
    parent = NULL, ...)

## S3 method for class 'guiWidgetsToolkittcltk'
.gbasicdialog(toolkit, title = "Dialog",
    parent = NULL, do.buttons = TRUE, handler = NULL, action = NULL, ...
)

## S3 method for class 'guiWidgetsToolkittcltk'
.galert(toolkit, msg, title = "message",
    delay = 3, parent = NULL, ...)

## S3 method for class 'guiWidgetsToolkittcltk'
.gmenu(toolkit, menu.list = list(),
```
popup = FALSE, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gaction(toolkit, label, tooltip = NULL,  
    icon = NULL, key.accel = NULL, handler = NULL, action = NULL,  
    parent = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gbutton(toolkit, text, handler, action,  
    container, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gcalendar(toolkit, text = "",  
    format = "%Y-%m-%d", handler = NULL, action = NULL,  
    container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
ghasBox(toolkit, text,  
    checked = FALSE, use.togglebutton = FALSE, handler = NULL,  
    action = NULL, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gtable(toolkit, items,  
    multiple = FALSE, chosen.col = 1, icon.col = NULL,  
    tooltip.col = NULL, handler = NULL, action = NULL,  
    container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
ghasBoxGroup(toolkit=NULL, items, checked = FALSE, horizontal = FALSE,  
    use.table = FALSE, handler = NULL, action = NULL, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gComboBox(toolkit, items, selected = 1,  
    editable = FALSE, coerce.with = NULL, handler = NULL,  
    action = NULL, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gdf(toolkit, items = NULL,  
    handler = NULL, action = NULL, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gedit(toolkit, text = "", width = 25,  
    coerce.with = NULL, initial.msg = initial.msg, handler = NULL,  
    action = NULL, container = NULL, ...)

## S3 method for class 'guiWidgetsToolkitctk'
gexpandGroup(toolkit, text, markup,  
    horizontal = TRUE, handler = NULL, action = NULL, ...)


---

# S3 method for class 'guiWidgetsToolkittcltk'
.gfile(toolkit, text = "", type = c("open", "save", "selectdir"), initial.filename = NULL, initial.dir = getwd(), filter = list(), multi = FALSE, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gfilebrowse(toolkit, text = "", type = c("open", "save", "selectdir"), initial.filename = NULL, initial.dir = getwd(), filter = list(), quote = TRUE, handler = NULL, action = NULL, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gformlayout(toolkit, align = "left", spacing = 5, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.ggroup(toolkit, horizontal = TRUE, spacing = 5, use.scrollwindow = FALSE, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gframe(toolkit, text, markup, pos, horizontal = TRUE, spacing = 5, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gimage(toolkit, filename = "", dirname = "", stock.id = NULL, size = ",", handler = NULL, action = NULL, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.glabel(toolkit, text = "", markup = FALSE, editable = FALSE, handler = NULL, action = NULL, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.glayout(toolkit, homogeneous = FALSE, spacing = 10, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gnotebook(toolkit, tab.pos = 3, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'
.gpanedgroup(toolkit, horizontal = TRUE, container = NULL, ...)

# S3 method for class 'guiWidgetsToolkittcltk'

---
### S3 method for class 'guiWidgetsToolkittcltk'
.gprogressbar(toolkit, value, container, ...

### S3 method for class 'guiWidgetsToolkittcltk'
.gradio(toolkit, items, selected = 1,
    horizontal = FALSE, handler = NULL, action = NULL,
    container = NULL, ...)

### S3 method for class 'guiWidgetsToolkittcltk'
.gseparator(toolkit, horizontal = TRUE,
    container = NULL, ...)

### S3 method for class 'guiWidgetsToolkittcltk'
.gradio(toolkit, items, selected = 1,
    horizontal = FALSE, handler = NULL, action = NULL,
    container = NULL, ...)

### S3 method for class 'guiWidgetsToolkittcltk'
.gseparator(toolkit, horizontal = TRUE,
    container = NULL, ...)

### S3 method for class 'guiWidgetsToolkittcltk'
.gtimer(toolkit, ms, FUN, data = NULL,
    one.shot = FALSE, start = TRUE)

### S3 method for class 'guiWidgetsToolkittcltk'
.gtree(toolkit, offspring = NULL,
    offspring.data = NULL, chosen.col = 1, offspring.col = 2,
    icon.col = NULL, tooltip.col = NULL, multiple = FALSE,
    handler = NULL, action = NULL, container = NULL, ...)

### S3 method for class 'guiWidgetsToolkittcltk'
.gvarbrowser(toolkit, handler = NULL,
    action = "summary", container = NULL, ...)
## Arguments

- **toolkit**
  - toolkit

- **msg**
  - Character. Message to display.

- **title**
  - Character. Title

- **icon**
  - What icon to show

- **parent**
  - Hint as to where to display

- **...**
  - Ignored

- **text**
  - Character. Initial text

- **do.buttons**
  - FALSE to suppress buttons when no parent

- **handler**
  - Handler called when Ok button invoked

- **action**
  - Passed to handler for OK button

- **delay**
  - Delay

- **menu.list**
  - A list defining the menu structure. Named sub lists determine the submenu titles and structure. The list may have components of class: GAction, mapped to a button; GSeparator, mapped to a horizontal separator; GRadio, mapped to linked buttons; or GCheckbox, mapped to a checkbox button.

- **popup**
  - Logical. If true, make a popup window to be added through a handler call

- **container**
  - A parent container. When a widget is created it can be incorporated into the widget heirarchy by passing in a parent container at construction time. (For some toolkits this is not optional, e.g. gWidgets2tcltk or gWidgets2WWW2.)

- **label**
  - Label for action

- **tooltip**
  - Toolktip for action

- **key.accel**
  - Keyboard accelerator. If given, parent must be specified.

- **format**
  - Date format
checked

use.togglebutton

Use a toggle button (shows depressed) not a check box

items
data.frame specifies items for selection. May be a vector, matrix or data frame

multiple

logical allow multiple selection

chosen.col

which value from the row is returned by selection

icon.col

NULL or integer. If latter, specifies column containing stock icon

tooltip.col

NULL or integer. If latter, specifies column containing tooltip

horizontal

logical. If true displayed horizontally, else vertically

use.table

logical. If supported, and TRUE then uses a table widget with scrollbars

selected

integer. Which item (by index) is selected. Use -1 for no selection

editable

logical. Is user allowed to edit value

coerce.with

A function of function name to be called before selected value is returned by

width

width of widget

initial.msg

initial.msg

markup

does label use markup (toolkit specific)

type

type of browser: to open a file, to save a file or to select a directory

initial.filename

Suggested file name

initial.dir

initial directory. If a filename is given, and is not an absolute name, this will be

filter

A filter specification. This can be a named character vector of file extensions or

multi

Logical. Allow multiple files to be selected?

quote

quote output

align

alignment of label. Left justify or center balance. Leave as "default" for underlying toolkit default.

spacing

spacing between columns
logical. Either TRUE, "TRUE", FALSE, "FALSE", "y", or "x". For all toolkits a
non-FALSE value will place the child components into a scrollable container.
For some toolkits this will only be in the direction of packing. If the toolkit
allows it (RGtk2), then values of "x" or "y" can be used to override the default
scrolling directions. A box container with scrollwindows should have it size set
either directly or through packing with expand=TRUE as its size request will not
reflect the size of its child components.

position of label: 0=left, 1=right, some toolkit allow values in between

basename of file
dirname of file
stock id of icon (if non NULL)
size when a stock id (toolkit dependent)
are cells all the same size
integer. Position of tabs, 1 on bottom, 2 left, 3 top, 4 right. (If supported)
a list or menu bar specifying the new menubar
If a number of length one then a starting point, in which case to, by are passed
to seq. Otherwise a sequence of values for which sort(unique(from)) will order
ending point when from is starting point
step size if not specified by from
digits
height of widget (when width is specified)
font attributes for text buffer. One can also specify font attributes for insertion.
The font attributes are specified with a list with named components, with names
and values coming from:

weight in c("light", "normal", "bold", "heavy")
style in c("normal", "oblique", "italic")
family in c("sans", "helvetica", "times", "monospace")
size in c("xx-small", "x-small", "small", "medium", "large", "x-large", "xx-
large")
foreground a value in colors()
background a value in colors()
logical do lines wrap
interval in milliseconds
FUnction to call. Has one argument, data passed in
passed to function
logical. If TRUE, called just once, else repeats
logical. If FALSE, started by start_timer OO method. (Call obj$start_time()).
function. A function passed values path and data, the latter from offspring$data.
The path is the current position of the parent item using the named keys from
the chosen column.
offspring.data Passed to second argument of offspring function. Used to parameterize a function call.

offspring.col integer or column name. Points to column containing logical values indicating if a row has offspring.

visible logical. If codeTRUE window is drawn when constructed. Otherwise, window can be drawn later using visible<-. This value can default to FALSE by setting the option: options("gWidgets::gwindow-default-visible-is-false"=TRUE). There are advantages: windows can draw slowly when adding many items. With gWidgets2RGtk2, the ggraphics widget can like to be added to an undrawn widget as this avoids sizing issue.

name name of icon

iconNames names of icons

iconFiles path of icons

obj obj to get icon from

See Also

The documentation for this is found at gbutton.

The documentation for this is found at gprogressbar.
Description

show the word list

Arguments

str  a string. If missing do nothing, otherwise match against string to generate word list. Popup menu depending on length

Methods

add_bindings()  Add bindings to the entry box
clear_error()  Clear error message
clear_init_txt(...)  clear out init text, set back to black
find_match(x)  Find match in word list
make_styles(bg = "#ff6622")  Create tcl styles, cf http://paste.tclers.tk/506
set_error(msg)  Add error state and message to widget
set_init_txt(msg)  set initial text, gray out
set_invalid(value, msg)  Set widget as invalid or not
set_validator(FUN)  Set a function to do the validation
validate_input()  Return logical indicating if input is valid

getWidget.tkwin  method for getWidget defined in gWidgets2

Description

method for getWidget defined in gWidgets2

Usage

## S3 method for class 'tkwin'
getWidget(obj)

Arguments

obj  object
GMenuBar  
Toplevel menu bar

Description

The GMenuBar class provides functionality for a top-level menubar. In tlctk the menubars can show either gaction items (proxied as buttons), gradio items or gcheckbox items. In the latter two cases, one uses the parent argument – not the container argument – to specify the parent container. Such items can also be shared with toolbars.

Usage

GMenuBar(...)  

Arguments

... passed to constructor

Examples

```r
## Not run:
w <- gwindow("having fun?")
 sb <- gstatusbar("Your message here...", cont=w)
g <- ggroup(cont=w, horizontal=FALSE)
f <- function(h,...) message(h$obj$getValue())

l <- list(file=gaction("file", handler=function(h,...) print("file"),
                  key.accel="<Control-x><Control-s>", parent=w),
     ok=gaction("ok", icon="ok", handler=function(h,...) print("ok")),
     radio=list(      
                   rb=gradio(state.name[1:3], parent=w, handler=function(h,...)
                                   print(h$obj$getValue()))
                     )
     sep=gseparator(vertical=TRUE),
     cb=gcheckbox("really", parent=w, handler=function(h,...) print(h$obj$getValue()))
                   )

mlist <- list(File=l)
mb <- gmenu(mlist, cont=w)

## End(Not run)
```
GSpinButton

Spinbutton class

Description

Spinbutton class

Usage

GSpinButton(...)

Arguments

... passed to constructor

GSpinButton is the base class for spin buttons. The widget is buggy in tcltk, atleast with the old non-themed style and a Mac running tcltk under X11. The bug is the thing keeps spinning when the buttons are pressed.

GStatusBar

GStatusBar is the base class for a status bar

Description

The GStatusBar class inherits for GBoxContainer meaning it can be used as a parent container. As such, one can add additional widgets beyond the plain label that is the main property of this widget.

Usage

GStatusBar(...)

Arguments

... passed to constructor
**Description**

The toolbar is a container, so can have other widgets added to it as though it were a box container. Buttons should be added as action items, so that they are rendered in the proper style. Check buttons should be given the argument `use.togglebutton`. Use `addSpring` to right align items.

**Arguments**

... passed to constructor

**Methods**

- `add_gaction_toolitem(obj)` Helper to add a gaction item
- `add_gseparator_toolitem()` Helper to add a separator
- `add_toolbar_items(items)` Map a toolbar list, a named list of gaction items or gsepartor items
- `clear_toolbar()` Clear toolbar items
- `get_widget()` What widget do we use for the parent of the children
- `set_value(value, ...)` We can’t really adjust spacing between children after they have been positioned.

---

**Description**

toolkit class for Tk

---

**Description**

toolkit class for Tk
**gwidgets2_tcltk_column_alignment**

*align a column based on the class of the content*

**Description**

Gives visual difference to data based on its class

**Usage**

```r
gwidgets2_tcltk_column_alignment(x)
```

```r
## Default S3 method:
gwidgets2_tcltk_column_alignment(x)
```

```r
## S3 method for class 'numeric'
gwidgets2_tcltk_column_alignment(x)
```

```r
## S3 method for class 'logical'
gwidgets2_tcltk_column_alignment(x)
```

**Arguments**

- `x` column data to align

**Value**

anchor string

---

**gwidgets2_tcltk_format_to_char**

*Format object to character class for inclusion in a table*

**Description**

Gives chance to do more than `as.character`

**Usage**

```r
gwidgets2_tcltk_format_to_char(x)
```

```r
## Default S3 method:
gwidgets2_tcltk_format_to_char(x)
```

```r
## S3 method for class 'factor'
gwidgets2_tcltk_format_to_char(x)
```
List-class

Arguments

x object to format to character class

Value

object of character class with possible formatting.

List-class

A list extension class.

Description

Like a list, but has some methods. Completely superflous, but makes copying some code algorithms easier. We implement methods such as append, push, pop and each for iteration. As well, there are some lookup methods.

Arguments

... passed to constructor

Methods

contains(name) TRUE if name is key in array
core() return list
each(FUN, ...) Iterator for lists, like sapply, but FUN gets passed index, key, and value
flush(...) Reset array, return contents as list
get_by_name(name) get item under name
get_id() Return an id, or name, for an object
get_item(index) Get item by index
insert(x, name, index) Insert item into List with 0 the head and index=len() the tail
makeCalendar

len() length
pluck(id, FUN, ...) Like ext.pluck. Returns array with 'id' extracted from each item in the List
pop() pop last element of list
push(x, name) Append x with optional name. If name not specified new id created. Returns name

makeCalendar

Description

make a calendar...

Usage

makeCalendar(date_var, widget, date, date_format = "%Y-%m-%d", set_value)

Arguments

date_var var
widget var
date var
date_format var
set_value var from chron with slight change to arguments
Index

*Topic **package**

gWidgets2tcltk-package, 2
.addStockIcons.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gaction.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.galert.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gbasicdialog.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gbutton.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gcalendar.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gcheckbox.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gcheckboxgroup.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gcombobox.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gconfirm.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gdf.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gedit.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.getStockIconByName.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.getStockIcons.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gexpandgroup.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gfile.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gfilebrowse.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gformlayout.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gframe.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.ggroup.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gimage.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.ginput.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.glableguiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.glayout.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gmenu.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
.gmessage.guiWidgetsToolkitTcltk
  (gmessage.guiWidgetsToolkitTcltk), 4
INDEX

...