Subject: cancel button to stop w/in an iterative loop?
Posted by T Bowers on Mon, 08 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Hi.
I'm trying to create a simple cancel button for my software.
I slapped a GUI on an iterative function that processes alot
of data files. I want a cancel button to appear just as the run
enters the processing loop, and exit out of processing if the
user pressed the button.
Below is a file tGUI.pro to show one of the variations I've
tried. I even tried to register the cancelBase base widget
with xmanager, then destroy it in the cancelProcessing()
callback, then check the value of xregistered in the outer
loop to see if it's registered. No such luck. I'm obviously
missing something here. Some of this code is strait outta
my app so please forgive my coding style (e.g. the loop
to create the widget buttons in the procedure tGUI). Oh,
and global variables via a COMMON block is my LEAST
favorite option. If that's the only way to go, I guess I'll have
to do it, but that'll be my last option. Even so, I did try it with a
common block, and I still couldn't get it to work.

Thanks alot in advance

IDL> print, !version
{ x86 Win32 Windows 5.2 Oct 30 1998}

```
;---CUT HERE-----
 ;////////////////////////////////////////////////////////// ////////////////
//////////////
function cancelProcessing_event, event
 widget_control, event.top, GET_VALUE=eventValue
 if eventValue EQ "CANCEL PROCESSING" then $
  widget_control, event.id, SET_UVALUE = 2
 return, 1
end
 ;////////////////////////////////////////////////////////// ////////////////
//////////////

 ;////////////////////////////////////////////////////////// ////////////////
//////////////
function processData
 ;//Create my cancel button
  cancelBase = Widget_Base(UNAME='cancelBase',/ALIGN_CENTER,$
   /GRID_LAYOUT ,TLB_FRAME_ATTR=9)
  cancelButton = Widget_Button(cancelBase, UNAME='cancelButton',$
   /ALIGN_CENTER ,VALUE='CANCEL PROCESSING', $
```

```
   EVENT_FUNC="cancelProcessing_event")
  Widget_Control, /REALIZE, cancelBase

 ;//Set something I can use to know it's been pressed
 widget_control, cancelBase, SET_UVALUE=0

 ;//Create an outer and inner loop. I wanna check to see if the cancel
 ;// button was pressed at the beginning of every iteration of outer loop.
  for j = 1,3 do begin
  widget_control, cancelBase, GET_UVALUE=widValue
  if  widValue EQ 2 then begin
   print, "User Cancelled"
   goto, BREAK
  endif
  for i = 0, 100 do begin
   print, i*j
  endfor
 endfor

 ;//User pressed cancel button so jumped to here
 BREAK:
 ;// ...destroy the cancel button 'n base
 widget_control, cancelBase, /destroy

 return, 0
end
 ;////////////////////////////////////////////////////////// /////////////////
//////////////

 ;////////////////////////////////////////////////////////// /////////////////
//////////////
pro mainForm_event, ev
 widget_control, ev.id, GET_VALUE = eventval
 case eventval of
  "Process Data": retCode = processData()
  "Quit": widget_control, ev.top, /destroy
 endcase
end
 ;////////////////////////////////////////////////////////// /////////////////
//////////////

 ;////////////////////////////////////////////////////////// /////////////////
//////////////
pro tGUI
 ;//There'll be 2 buttons
 mainFormButtonList = ["Process Data","Quit"]
 ;//Create base
 baseWidget = widget_base(TLB_FRAME_ATTR=1)
```

```
  ;//Create buttton base widget off of base
 buttonBaseWidget = widget_base(baseWidget, /column, space=10)

 ;//create buttons...
 numButtons = n_elements(mainFormButtonList)
 mainFormButtons = lonarr(numButtons)
 for i=0,numButtons-1 do begin
  mainFormButtons[i] = widget_button(buttonBaseWidget,
value=mainFormButtonList[i])
 endfor

 widget_control, baseWidget, /realize

 ;//Pass off to xmanager
 xmanager, "mainForm", baseWidget, /no_block
end
 ;////////////////////////////////////////////////////////// /////////////////
//////////////
;---CUT HERE-----
```