
Subject: Re: pulldown menu with multiple selections - Yes we can!

Posted by [Michael Asten](#) on Fri, 29 Oct 1999 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

pei zeng wrote:

> Hi, Guys
>
> Does anybody know how to make a pulldown menu with multiple selections?
>
> Thanks!
>
> pei

I did it using the pull down menu generated by the idl5.2 GuiBuilder,
with
subsequent addition to code to extract the "multiple" selection.
Attached is a demo example - the task is to select multiple 'loops'
from
the left column of a widget, and single 'line' and 'component' from 2nd
and
3rd columns of the widget.

On win95/98/NT systems, multiple menu item selection is by the windows
convention of pressing <control> rightMouseClicked to select individual
items
additively, or pressing <shift> rightMouseClicked to select all menu items
between two highlighted items. I assume Mac and Unix have mouse
conventions for equivalent processes.

I acknowledge the help of RSI Tech support in getting this multiple menu
item selection working - it is not well documented in the idl5.2 online
help.

Regards,
Michael Asten

Code follows:

```
;
;
; IDL Widget Interface Procedures. This Code is automatically
; generated and should not be modified.
;
;
; Generated on: 12/10/98 13:43.49
;
;
PRO WID_BASE_0_event, anEvent
common
```

```

select_subs,loop_index,line_index,comp_index,freq_index,gain_index

_widID_ = anEvent.top

CASE anEvent.id OF

WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_LIST_3'): BEGIN
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_LIST' then $
        Freq_index= anEvent.index
    END
WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_LIST_4'): BEGIN
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_LIST' then $
        Gain_index= anEvent.index
    END
WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_LIST_1'): BEGIN
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_LIST' then $
        Line_index= anEvent.index
    END
WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_LIST_2'): BEGIN
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_LIST' then $
        Comp_index= anEvent.index
    END
WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_LIST_0'): BEGIN

; PROCESSING OF LIST SELECTION FOR LEFT COLUMN,
; WHERE MULTIPLE SELECTION REQUIRED
; need a second call to widget_info, to save name of wid
wid = widget_info(_widID_, FIND_BY_UNAME='WID_LIST_0')
; wid now contains the widget id of WID_LIST_0
print,'wid = ', wid
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_LIST' then begin
        ;now get the indices of the selected list items
        ; this is a scalar, if only 1 choice made
        ; this is an array if multiple choices selected
        loop_index=widget_info(wid,/list_select)
            ;print,'anEvent.index=',anEvent.index
            ;loop_index= anEvent.index
        print,'loop_index from multiple select=',loop_index
    endif
END

WIDGET_INFO(_widID_, FIND_BY_UNAME='WID_BUTTON_0'): BEGIN
    if TAG_NAMES(anEvent, /STRUCTURE_NAME) EQ 'WIDGET_BUTTON' then $
        widget_control(anEvent.top,/destroy)
    END
ELSE:
ENDCASE

```

END

PRO WID_BASE_0, GROUP_LEADER=_VWBGroup_, _EXTRA=_VWBExtra_
,loops,lines,comps,freqs,gains

;delete this - not needed for simple demo program
;RESOLVE_ROUTINE, 'select_subset2_eventcb' ; Load event callback
routines

WID_BASE_0 = WIDGET_BASE(GROUP_LEADER=_VWBGroup_, \$
UNAME='WID_BASE_0',SCR_XSIZE=491,SCR_YSIZE=175,TITLE='IDL' \$
,SPACE=3,XPAD=3,YPAD=3)

WID_LIST_2 = WIDGET_LIST(WID_BASE_0, UNAME='WID_LIST_2' \$
,XOFFSET=180,YOFFSET=43,SCR_XSIZE=57,SCR_YSIZE=78,value=comps)

WID_LIST_3 = WIDGET_LIST(WID_BASE_0, UNAME='WID_LIST_3' \$
,XOFFSET=254,YOFFSET=43,SCR_XSIZE=57,SCR_YSIZE=78,value=freqs)

WID_LIST_4 = WIDGET_LIST(WID_BASE_0, UNAME='WID_LIST_4' \$
,XOFFSET=337,YOFFSET=43,SCR_XSIZE=57,SCR_YSIZE=78,value=gains)

WID_LABEL_0 = WIDGET_LABEL(WID_BASE_0, UNAME='WID_LABEL_0' \$
,XOFFSET=38,YOFFSET=20,/ALIGN_LEFT,VALUE='LOOP')

WID_LABEL_4 = WIDGET_LABEL(WID_BASE_0, UNAME='WID_LABEL_4' \$
,XOFFSET=365,YOFFSET=20,SCR_XSIZE=28,SCR_YSIZE=18 \$
,/ALIGN_LEFT,VALUE='GAIN')

WID_LABEL_3 = WIDGET_LABEL(WID_BASE_0, UNAME='WID_LABEL_3' \$
,XOFFSET=276,YOFFSET=20,/ALIGN_LEFT,VALUE='FREQ')

WID_LABEL_1 = WIDGET_LABEL(WID_BASE_0, UNAME='WID_LABEL_1' \$
,XOFFSET=100,YOFFSET=20,SCR_XSIZE=62,SCR_YSIZE=18 \$
,/ALIGN_LEFT,VALUE='LINE/HOLE')

WID_LIST_1 = WIDGET_LIST(WID_BASE_0, UNAME='WID_LIST_1' \$
,XOFFSET=103,YOFFSET=43,SCR_XSIZE=57,SCR_YSIZE=78,value=lines)

```

WID_LABEL_2 = WIDGET_LABEL(WID_BASE_0, UNAME='WID_LABEL_2' $
,XOFFSET=205 ,YOFFSET=20 ,/ALIGN_LEFT ,VALUE='COMPNT')

WID_BUTTON_0 = WIDGET_BUTTON(WID_BASE_0, UNAME='WID_BUTTON_0' $
,XOFFSET=418 ,YOFFSET=43 ,/ALIGN_CENTER ,VALUE='APPLY')

WID_LIST_0 = WIDGET_LIST(WID_BASE_0, UNAME='WID_LIST_0' ,XOFFSET=18 $
,YOFFSET=44 ,SCR_XSIZE=43 ,SCR_YSIZE=78 ,/multiple,value=loops)

WIDGET_CONTROL, /REALIZE, WID_BASE_0

XMANAGER, 'WID_BASE_0', WID_BASE_0

END
;
; Empty stub procedure used for autoloading.
;
PRO select_subset2, GROUP_LEADER=_VWBGroup_, _EXTRA=_VWBExtra_
,loops,lines,comps,freqs,gains
  WID_BASE_0, GROUP_LEADER=_VWBGroup_, _EXTRA=_VWBExtra_
,loops,lines,comps,freqs,gains
END

; MAIN PROG
common
  select_subs,loop_index,line_index,comp_index,freq_index,gain_index
loops=['loop1','loop2','loop3','loop4','loop5']
lines=['line1','line2']
comps=['Z']
freqs=['P6.25']
gains=['0']
loop_index=0 & line_index=0 & comp_index=0 & freq_index=0 & gain_index=0
select_subset2 ,loops,lines,comps,freqs,gains
  print,loop_index,line_index,comp_index,freq_index,gain_index
end

```
