
Subject: Re: Help with pick menu widget code
Posted by [davidf](#) on Thu, 17 Sep 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Rose Longfield (rmlongfield@my-dejanews.com) writes:

> Does this mean that I have to know before-hand that my module is going to
> cause an error? (Isn't this rather pessimistic) Or, I guess it is a good
> de-bugging technique, to be applied AFTER the program crashes. In this case I
> will try it out.

I use it as a debugging technique after I get over the shock
and dismay that my carefully crafted program didn't run perfectly. :-)

> I haven't had much luck with error handling in IDL. It usually causes me more
> trouble because I don't notice that an error has occurred.

This can be one of the big disadvantages of the CATCH mechanism
for catching errors. I once spent about three hours thinking IDL
had gone completely bonkers only to finally realize I had added
a silent CATCH to one of my modules. Needless to say, I don't
add any more silent CATCHes to **anything**. I write them like this:

```
CATCH, error
IF error NE 0 THEN BEGIN
    Catch, /Cancel
    ok = Dialog_Message(!Error_State.Msg)
    Print, !Error_State.Msg
    Widget_Control, event.top, Set_UValue=info, /No_Copy
    RETURN
ENDIF
```

The Cancelling of the Catch keeps me from the infinite loop
that results when I incorporate errors in my error handling
code. Sigh... The Dialog_Message makes sure I know an error
occurred, because it is a modal dialog I **must** respond to.
The Print statement is there for me to refer to when I hit
the OK on the dialog before carefully reading the message
and I am trying to debug my code.

This CATCH will keep a widget program alive indefinitely,
but it also usually gives me enough notice to be able
to fix a problem. I suppose more bells and whistles and
flashing lights wouldn't hurt. My problem is that
I see this damn dialog so often that I have become
habituated to it and almost never **really** see it. :-)

Of course, occasionally, I want to disable this CATCH,

perhaps to find one of those "client application" errors.
Rather than add semi-colons to each of these lines (there are occasionally more as I sometimes do *real* error handling too), I modify the code like this:

```
;CATCH, error
error = 0
IF error NE 0 THEN BEGIN
    Catch, /Cancel
    ok = Dialog_Message(!Error_State.Msg)
    Print, !Error_State.Msg
    Widget_Control, event.top, Set_UValue=info, /No_Copy
    RETURN
ENDIF
```

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
