Subject: Re: Oject programming error in non-object program
Posted by Bill Nel on Thu, 21 Jun 2012 17:34:12 GMT

On Jun 21, 1:28 pm, David Klassen <klas...@rowan.edu> wrote:
> This is frustrating me because my IDL skills are at the IDL 5.0 level, but since ITT won't allow me to simply move software, I have to re-buy each time I get a new machine, I have IDL 8...
>
> The program is a massive thing and mostly works fine but every now and again (and I can't find a reason---but it seems to happen if I recompile the program) I get this error:
>
> % Object reference expression not allowed in this context: <OBJREF
>   (<ObjHeapVar16(LIST)>)>.
>
> and I have no clue what it means. The program was written in IDL 3, and last updated under IDL 5 so there are no object oriented programming statements.
>
> The crash happens on:
>   for ii=0l,n_list-1 do oplot,$
>   [new_vecs(list(ii),i)],[new_vecs(list(ii),j)],$
>   psym=pssym,color=k_col(list(ii))

LIST is a function since IDL 8.0 -- you need to rename your LIST array
variable. (Using square brackets to index arrays instead of
parenthesis might help temporarily, but it's probably not a great
solution.)

Subject: Re: Oject programming error in non-object program
Posted by David Fanning on Thu, 21 Jun 2012 17:44:00 GMT

David Klassen writes:

> Any ideas on what the clash might be?!

You can add your "list" variable to a long list (pun intended)
of variable and function names that have been appropriated
by IDL 8. :-)

  http://www.idlcoyote.com/ng_tips/imgvar.php
  http://www.idlcoyote.comng_tips/idl8_name_conflicts.html

Cheers,

David

--
David Fanning, Ph.D.

Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Oject programming error in non-object program
Posted by David Klassen on Thu, 21 Jun 2012 18:29:41 GMT

On Thursday, June 21, 2012 1:34:12 PM UTC-4, Bill Nel wrote:
> On Jun 21, 1:28 pm, David Klassen <klas...@rowan.edu> wrote:
>> This is frustrating me because my IDL skills are at the IDL 5.0 level, but since ITT won't allow me to simply move software, I have to re-buy each time I get a new machine, I have IDL 8...
>>
>> The program is a massive thing and mostly works fine but every now and again (and I can't find a reason---but it seems to happen if I recompile the program) I get this error:
>>
>> % Object reference expression not allowed in this context: <OBJREF
>>   (<ObjHeapVar16(LIST)>)>.
>>
>> and I have no clue what it means. The program was written in IDL 3, and last updated under IDL 5 so there are no object oriented programming statements.
>>
>> The crash happens on:
>>   for ii=0l,n_list-1 do oplot,$
>>   [new_vecs(list(ii),i)],[new_vecs(list(ii),j)],$
>>   psym=pssym,color=k_col(list(ii))
>
> LIST is a function since IDL 8.0 -- you need to rename your LIST array
> variable. (Using square brackets to index arrays instead of
> parenthesis might help temporarily, but it's probably not a great
> solution.)

Thank you! (and D Fanning)

This was driving me nuts. Odd thing: it works *most* of the time...

Ugh; so now I've got to comb through a few thousand lines of code (well, OK, global search and replace—and hope that doesn't break something else) and change my variable name.

Subject: Re: Oject programming error in non-object program
Posted by David Klassen on Thu, 21 Jun 2012 19:35:09 GMT

On Thursday, June 21, 2012 1:44:00 PM UTC-4, David Fanning wrote:
> David Klassen writes:

>
>> Any ideas on what the clash might be?!
>
> You can add your "list" variable to a long list (pun intended)
> of variable and function names that have been appropriated
> by IDL 8. :-)
>
>   http://www.idlcoyote.com/ng_tips/imgvar.php
>   http://www.idlcoyote.comng_tips/idl8_name_conflicts.html
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Thanks for those articles. I see I need to surf by more often.

And good thing I never got in the habit of using image as a variable. I'm lazy and just use img.
But I have seen the issue with legend, so I've just pre-compiled it by full path in my start-up file
(which I see was their solution).

I guess the best thing going forward may be to "personalize" all our own libraries of routines (and
variables?!).