Subject: Re: A simple IF statement question
Posted by Michael Wallace on Mon, 14 Feb 2005 21:09:02 GMT
View Forum Message <> Reply to Message

>>> I am using IDL 6.1 on Linux SUSE. I am writing a simple code using the
>>> IF statement and am wondering about the following result:
>>>
>>> IDL> IF 1 THEN PRINT, '1' ELSE PRINT, 'None'
>>> 1
>>> IDL> IF 2 THEN PRINT, '2' ELSE PRINT, 'None'
>>> None
>>> IDL> IF 19 THEN PRINT, '19' ELSE PRINT, 'None'
>>> 19
>>> IDL> IF 24 THEN PRINT, '24' ELSE PRINT, 'None'
>>> None
>>> IDL> IF 0 THEN PRINT, '0' ELSE PRINT, 'None'
>>> None
>>>
>>> Am I wrong when I expect the IF statement to return always TRUE if the
>>> condition is not 0 (I mean something like 1,2,3,4,....)?
>>
>> Yes, you are wrong. :-)
>>
>> Here is an article you might want to read:
>>
>> http://www.dfanning.com/code_tips/bitwiselogical.html
>
>
> It has been pointed out to me that the article is a bit
> deficient in that it doesn't mention the LOGICAL_PREDICATE
> compiler option. If you set:
>
> COMPILE_OPT LOGICAL_PREDICATE
>
> Then 0 is FALSE and everything else is TRUE. That probably
> makes more sense to *everyone*! :-)

Nice article, but it doesn't show the logical operators &&, || and ~.
I'm not complaining, but thought that if you're ever bored one day, you
can explore the facets of ~ and NOT and AND and && or OR and ||.
Actually, I only mention that because I think it's really cool to have
part of an actual sentence that reads aloud as "not and not and and and
and or or and or." And then there's also LOGICAL_OR and LOGICAL_AND
operators. Too many operators!!

Seriously now, using logical not (~) you can achieve the same thing as
the LOGICAL_PREDICATE. Actually ~ is tied to the same definition as

LOGICAL_PREDICATE, so it behaves the same way. If you want a variable var to evaluate to false when 0 and true otherwise, all you need is ~(~var). If you just did ~var, you'd get the exact opposite of what we want -- 0 is TRUE (1) and everything else is FALSE (0). The second logical not flips this result. Just another way to skin a cat.

-Mike
