Subject: Re: string conversion & integer
Posted by rkombiyil on Thu, 08 Nov 2007 23:27:07 GMT
View Forum Message <> Reply to Message

On Nov 9, 2:37 am, David Fanning <da...@dfanning.com> wrote:
> metachronist writes:
>> Okay folks, I am sure this has been discussed many times already.
>> Maybe trivial/silly... Anyway, to make sure:
>
>> Problem:
>> ------------
>>> print,where(X.dd eq dd_d)
>>   -1
>> I think it's got to do with internal representation:
>
>> dd_d is originally read in as "string", for ease of concatenation and
>> constructing filename, along with year and month etc. Say, for
>> example, as '07'
>
>> But field "dd" in X.dd is defined in the format statement as:
>> dd:0L , say for example, from 1 to 10.
>> So there is a type mismatch. I tried:
>
>>> print, where(X.dd eq fix(dd_d,type=3)
>>   -1
>>> print, where(X.dd eq long(dd_d))
>>   -1
>
>> No luck. Is there a way out "other than defining dd_d as long
>> originally and make excessive use of string/strtrim/strmid etc while
>> constructing filenames?
>
> Well, here is an experiment:
>
> IDL> x = {dd:indgen(10)}
> IDL> y = '8'
> IDL> print, where(x.dd eq fix(y))
>   8
>
> I think maybe you are making assumptions that just
> ain't true in your code. :-)
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:http://www.dfanning.com/

Thank you for your reply. I am glad the reply was gentle ;-)

This is really weird! And I can't figure out what might be wrong. I
pass dd_d via call to a procedure, wherein it gets stored as e_dd. It
is getting passed alright, but I try to do the conversion within the
procedure and it fails. Am I doing something so fundamentally wrong
here?! Please see:

```
;----------  ---------------
pro test,e_dd
print,e_dd,size(e_dd,/type)
 x={dd:[1l,1l,2l,2l,4l,4l,4l,5l,5l,6l,6l,7l,7l,7l,8l,8l,8l,9l ,9l]}
print,where(x.dd eq fix(e_dd))
end

close,/all
numdays=''
openr,ilun,'test.dat',/get_lun
readf,ilun,numdays
dd_d=strarr(numdays)
readf,ilun,dd_d
free_lun,ilun
test,dd_d
end
;-------------[/sample_code]-------------
```
=================================
and 'test.dat' has 2 lines:
1
07
=================================
I am wondering why the above doesn't work? Still returns -1, whereas I
"would expect" that it will return locations of 7's in the array ?!! :-
(Any thoughts?
Thanks,
~r