
Subject: Re: Converting Doubles to Strings
Posted by [R.Bauer](#) on Tue, 08 Nov 2005 06:42:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

biophys wrote:

Don't do that conversion of an input float to double please use an internal variable and look at this nice article

http://www.dfanning.com/math_tips/double.html

cheers
Reimar

```
> Hi, David
>
> I think I've got a more decent solution by assuming that double
> precision numbers have exactly 14 significant digits and throw away
> zeros at the end. In IDL help, it says that double type has
> appoximately 14 digits of significance. So I am sure my solution will
> fail somewhere. However, so far it works pretty nice. Please let me
> know if any bugs should be fixed or any improvement should be made. A
> few examples here,
>
> IDL> print,dbl2str(-22.12)
> -22.12E
> IDL> print,dbl2str(-22.12d)
> -22.12D
> IDL> print,dbl2str(-22.1234567890)
> -22.12346E
> IDL> print,dbl2str(-22.1234567890d)
> -22.123456789D
> IDL> print,dbl2str(!dpi*1e20)
> 3.1415927165501E+20
> IDL> print,dbl2str(!dpi*1d20)
> 3.1415926535897D+20
>
> here's the code,
>
> ;converting float/double type numer into its original inputform
> function dbl2str, a
>
> tp=size(a,/type)
> if tp ne 4 and tp ne 5 then begin
>   a=double(a)
>   tp=5
> endif
```

```

> tps=tp eq 4?'E':'D'
>
> rawstr=strtrim(string(a,format='(g)'),2);full width G format
> sign=strmid(rawstr,0,1) eq '-'?'-':'';extract sign
> rawstr=sign eq ""?rawstr:strmid(rawstr,1);throw away sign
> epos=strpos(rawstr,'e')
> indx=epos gt -1?strmid(rawstr,epos+1):";extract exponent index
> rawstr=indx eq ""?rawstr:strmid(rawstr,0,epos);throw away exponent part
> dpos=strpos(rawstr,'. ');extract decimal point position
>
>
> ;rounding process(assume 14 significant digits)
> outstr=strarr(15)
> for i=0,14 do outstr[i]=strmid(rawstr,i,1)
> aux=fix(strmid(rawstr,16,1)) ge 5?1:0
> for i=14, 0, -1 do begin
>   if i ne dpos then begin
>     sumstr=strtrim(string(aux+fix(outstr[i])),2)
>     sumlen=strlen(sumstr)
>     outstr[i]=sumstr[sumlen-1]
>     aux=sumlen eq 1?0:1
>   endif
> endfor
>
> ;throw away '0's at the end
> ii=14
> while outstr[ii] eq '0' do begin
>   ii=ii-1
> endwhile
>
> ;reconstruct the inputform
> saux=aux ne 0?'1':"
> outstr=sign+saux+strjoin(outstr[0:ii])+tps+indx
>
> return,outstr
>
> end
>
> cheers,
> bp
>
> David Fanning wrote:
>> Folks,
>>
>> I've run into an interesting problem. I have a double precision
>> number that I wish to convert to a string (so I can put it into
>> a text widget, for example). I don't know ahead of time how many
>> significant digits will be in this number. The number could

```

```
>> look like this 45.6, or this 123456789.123456789, or this
>> -22.1234567890. If it looks like the latter number, I am
>> having a very hard time converting it to a string. For example,
>> this doesn't work:
>>
>> IDL> v = String(-22.1234567890, Format='(D0)')
>> IDL> Print, v
>>     -22.123457
>>
>> Is this a bug in IDL? Or am I overlooking something?
>>
>> Cheers,
>>
>> David
>> --
>> David Fanning, Ph.D.,
>> Fanning Software Consulting, Inc.
>> Coyote's Guide to IDL Programming: http://www.dfanning.com/
```
