Subject: Reading formatted input with READS
Posted by Stefano Garcia on Thu, 01 Sep 2016 03:27:14 GMT
View Forum Message <> Reply to Message

Hello everybody,

This is my first post so please let me know if I posted it in the wrong place.

My problem is purely that I am not yet familiarized with the IDL formats. I would like to use READS to read what is in between the square brackets below:

%M.z 20.37 [~] D 2011A&A...527A..78F
%M.z 20.37 [0.12] D 2011A&A...527A..78F

So, there is two cases, one when it is a STRING and the other, when it is a FLOAT.

Is there a way to read that?

I will appreciate any help.

Thanks

Subject: Re: Reading formatted input with READS
Posted by Helder Marchetto on Thu, 01 Sep 2016 08:50:06 GMT
View Forum Message <> Reply to Message

On Thursday, September 1, 2016 at 5:27:19 AM UTC+2, Stefano Garcia wrote:
> Hello everybody,
>
> This is my first post so please let me know if I posted it in the wrong place.
>
> My problem is purely that I am not yet familiarized with the IDL formats. I would like to use READS to read what is in between the square brackets below:
>
> %M.z 20.37 [~] D 2011A&A...527A..78F
> %M.z 20.37 [0.12] D 2011A&A...527A..78F
>
> So, there is two cases, one when it is a STRING and the other, when it is a FLOAT.
>
> Is there a way to read that?
>
> I will appreciate any help.
>
> Thanks

Hi,
dunno how to use reads, but I do these sort of stuff with string functions. Not as elegant, but

works. To check if the string between square brackets is a number or not, I use the isnumber() from Johns Hopkins University/Applied Physics Laboratory. You don't need the whole library. You can find it here: http://fermi.jhuapl.edu/s1r/idl/s1rlib/local_idl.html

```
IDL> str = '%M.z 20.37 [~]  D 2011A&A...527A..78F'
IDL> posStart = strpos(str, '[')+1
IDL> posEnd = strpos(str, ']')
IDL> print, strmid(str, posStart, posEnd-posStart)
~
IDL> print, isnumber(strmid(str, posStart, posEnd-posStart), value) gt 0
   0
IDL>
IDL> str = '%M.z 20.37 [0.12]  D 2011A&A...527A..78F'
IDL> posStart = strpos(str, '[')+1
IDL> posEnd = strpos(str, ']')
IDL> print, strmid(str, posStart, posEnd-posStart)
0.12
IDL> print, isnumber(strmid(str, posStart, posEnd-posStart), value) gt 0
```

I hope it helps.
Cheers,
Helder

PS: this is the right place for these questions

Subject: Re: Reading formatted input with READS
Posted by Markus Schmassmann on Thu, 01 Sep 2016 08:55:22 GMT
View Forum Message <> Reply to Message

On 09/01/2016 05:27 AM, Stefano Garcia wrote:
> This is my first post so please let me know if I posted it in the
> wrong place.
>
> My problem is purely that I am not yet familiarized with the IDL
> formats. I would like to use READS to read what is in between the
> square brackets below:
>
> %M.z 20.37 [~] D 2011A&A...527A..78F
> %M.z 20.37 [0.12] D 2011A&A...527A..78F
>
> So, there is two cases, one when it is a STRING and the other, when
> it is a FLOAT.
>
> Is there a way to read that?
Hallo Stefano, welcome

one way to read the string is the following:

```
 str=strMid(string,strPos(string,'[')+1,strPos(string,']')-st rPos(string,'[')-1)
 fltRegEx='^[-+]?(([0-9]*\.?[0-9]+([ed][-+]?[0-9]+)?)|(inf)|( nan))$'
isFloat=stRegEx(str,fltRegEx,/boolean,/fold_case)
if isFloat then out=float(str) else out=str
```

if you want to do it with READS, you should first figure out which terms
in your table have a fixed with, this will simplify considerably coming
up with a correct format code. For details on these, see:

http://www.harrisgeospatial.com/docs/format_codes.html

Markus

Subject: Re: Reading formatted input with READS
Posted by Markus Schmassmann on Thu, 01 Sep 2016 14:15:56 GMT
View Forum Message <> Reply to Message

On 09/01/2016 10:55 AM, Markus Schmassmann wrote:
> On 09/01/2016 05:27 AM, Stefano Garcia wrote:
>> This is my first post so please let me know if I posted it in the
>> wrong place.
>>
>> My problem is purely that I am not yet familiarized with the IDL
>> formats. I would like to use READS to read what is in between the
>> square brackets below:
>>
>> %M.z 20.37 [~] D 2011A&A...527A..78F
>> %M.z 20.37 [0.12] D 2011A&A...527A..78F
>>
>> So, there is two cases, one when it is a STRING and the other, when
>> it is a FLOAT.
>>
>> Is there a way to read that?
> one way to read the string is the following:
>
> str=strMid(string,strPos(string,'[')+1,strPos(string,']')-st rPos(string,'[')-1)
>
> fltRegEx='^[-+]?(([0-9]*\.?[0-9]+([ed][-+]?[0-9]+)?)|(inf)|(nan))$'
> isFloat=stRegEx(str,fltRegEx,/boolean,/fold_case)
> if isFloat then out=float(str) else out=str
>
> if you want to do it with READS, you should first figure out which terms
> in your table have a fixed with, this will simplify considerably coming
> up with a correct format code. For details on these, see:
>
> http://www.harrisgeospatial.com/docs/format_codes.html

made an error with the regex, hope this one is correct now:

fltRegEx='^[-+]?(((([0-9]*\.?[0-9]+)|([0-9]+\.))([ed]([-+]?[0-9]+)?)?)|(inf)|(nan))$'

Subject: Re: Reading formatted input with READS
Posted by Stefano Garcia on Thu, 01 Sep 2016 21:12:17 GMT
View Forum Message <> Reply to Message

This is indeed a very useful place.

Helder, thanks for the ISNUMBER functions like. I do not have yet the library you mentioned but I found something similar in the IDL Astronomy Users Library that I already have.

Markus, thanks for the usage of regular expresions, it was educative. Indeed, that was the original direction of my question.

Finally, I will do as follow

```
str = strsplit(str, "[]", /Extract)
str = str[1]
```

And then to check with one of those ISNUMBER functions like.

Thanks again.
