
Subject: Print 2 arrays side by side in one file
Posted by [a.mozafari1211](#) on Thu, 19 Nov 2009 13:34:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi;
I need to print 2 arrays side by side in one file I put something like
this:

```
openw,1,'FINAL.dat'  
printf,2, x,y, format='(3f14)'  
close,2
```

Unfortunately in output file it prints first x array and then y like
this:

```
x  
y  
will you please help me to fix this?
```

A.

Subject: Re: Pri
Posted by [KRDean](#) on Thu, 25 Feb 2010 15:38:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Feb 25, 7:53 am, Lasse Clausen <l...@lbnc.de> wrote:

```
> Hi there,  
>  
> I have written a little routine that prints out the usage of routines/  
> procedures written in IDL on the command line. Like so:  
>  
> IDL> pri, 'reverse'  
> +---+  
> Function REVERSE  
> Usage:  
>   Result = REVERSE(A, SUBSCRIPTIN, OVERWRITE=OVERWRITE)  
> Source:  
>   /usr/local/itt/idl/lib/reverse.pro  
> +---+  
>  
> or  
>  
> IDL> pri, 'pri'  
> +---+  
> Procedure PRI  
> Usage:  
>   PRI, ROUTINE_NAME, NO_COMPILE=NO_COMPILE, SOURCE_FILE=SOURCE_FILE,  
>   USAGE=USAGE
```

> Source:
> /davit/lib/vt/idl/misc/pri.pro
> +---+
>
> I find this routine incredibly helpful as it allows me to check how to
> use routines without having to leave the IDL command line.
>
> If you are interested, you can download it here:<http://sd-work5.ece.vt.edu/pri.pro>
>
> Cheers
> Lasse Clausen

Interesting! However, do I need PRINFO for it to work?

Kelly Dean
Fort Collins, CO

Subject: Re: Pri
Posted by [Lasse Clausen](#) on Thu, 25 Feb 2010 15:55:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Feb 25, 10:38 am, kBob <krd...@gmail.com> wrote:

> On Feb 25, 7:53 am, Lasse Clausen <l...@lbnc.de> wrote:

>
>
>
>> Hi there,
>
>> I have written a little routine that prints out the usage of routines/
>> procedures written in IDL on the command line. Like so:
>
>> IDL> pri, 'reverse'
>> +---+
>> Function REVERSE
>> Usage:
>> Result = REVERSE(A, SUBSCRIPTIN, OVERWRITE=OVERWRITE)
>> Source:
>> /usr/local/itt/idl/lib/reverse.pro
>> +---+
>
>> or
>
>> IDL> pri, 'pri'
>> +---+
>> Procedure PRI
>> Usage:
>> PRI, ROUTINE_NAME, NO_COMPILE=NO_COMPILE, SOURCE_FILE=SOURCE_FILE,

>> USAGE=USAGE
>> Source:
>> /davit/lib/vt/idl/misc/pri.pro
>> +---+
>
>> I find this routine incredibly helpful as it allows me to check how to
>> use routines without having to leave the IDL command line.
>
>> If you are interested, you can download it here:<http://sd-work5.ece.vt.edu/pri.pro>
>
>> Cheers
>> Lasse Clausen
>
> Interesting! However, do I need PRINFO for it to work?
>
> Kelly Dean
> Fort Collins, CO

Ah. No, not really, just replace the call to PRINFO with PRINT.

Lasse Clausen

Subject: Re: Pri
Posted by [mankoff](#) on Thu, 25 Feb 2010 16:11:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Feb 25, 6:53 am, Lasse Clausen <l...@lbnc.de> wrote:

> Hi there,
>
> I have written a little routine that prints out the usage of routines/
> procedures written in IDL on the command line. Like so:
>
> IDL> pri, 'reverse'
> +---+
> Function REVERSE
> Usage:
> Result = REVERSE(A, SUBSCRIPTIN, OVERWRITE=OVERWRITE)
> Source:
> /usr/local/itt/idl/lib/reverse.pro
> +---+
>
> or
>
> IDL> pri, 'pri'
> +---+
> Procedure PRI
> Usage:

> PRI, ROUTINE_NAME, NO_COMPILE=NO_COMPILE, SOURCE_FILE=SOURCE_FILE,
> USAGE=USAGE
> Source:
> /davit/lib/vt/idl/misc/pri.pro
> +---+
>
> I find this routine incredibly helpful as it allows me to check how to
> use routines without having to leave the IDL command line.
>
> If you are interested, you can download it here:<http://sd-work5.ece.vt.edu/pri.pro>
>
> Cheers
> Lasse Clausen

Looks useful, but similar to DOC_LIBRARY?

-k.

Subject: Re: Pri

Posted by [Lasse Clausen](#) on Thu, 25 Feb 2010 16:31:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Feb 25, 11:11 am, mankoff <mank...@gmail.com> wrote:

> On Feb 25, 6:53 am, Lasse Clausen <l...@lbnc.de> wrote:

>

>

>

>> Hi there,

>

>> I have written a little routine that prints out the usage of routines/

>> procedures written in IDL on the command line. Like so:

>

>> IDL> pri, 'reverse'

>> +---+

>> Function REVERSE

>> Usage:

>> Result = REVERSE(A, SUBSCRIPTIN, OVERWRITE=OVERWRITE)

>> Source:

>> /usr/local/itt/idl/lib/reverse.pro

>> +---+

>

>> or

>

>> IDL> pri, 'pri'

>> +---+

>> Procedure PRI

>> Usage:

```
>> PRI, ROUTINE_NAME, NO_COMPILE=NO_COMPILE, SOURCE_FILE=SOURCE_FILE,
>> USAGE=USAGE
>> Source:
>> /davit/lib/vt/idl/misc/pri.pro
>> +----+
>
>> I find this routine incredibly helpful as it allows me to check how to
>> use routines without having to leave the IDL command line.
>
>> If you are interested, you can download it here:http://sd-work5.ece.vt.edu/pri.pro
>
>> Cheers
>> Lasse Clausen
>
> Looks useful, but similar to DOC_LIBRARY?
>
> -k.
```

I wasn't aware of DOC_LIBRARY. You learn something every day...

And yes, I guess it is very similar to DOC_LIBRARY.

Lasse

Subject: Re: Pri
Posted by [JDS](#) on Thu, 25 Feb 2010 20:12:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Feb 25, 9:53 am, Lasse Clausen <...@lbnc.de> wrote:

```
> Hi there,
>
> I have written a little routine that prints out the usage of routines/
> procedures written in IDL on the command line. Like so:
>
> IDL> pri, 'reverse'
> +----+
> Function REVERSE
> Usage:
> Result = REVERSE(A, SUBSCRIPTIN, OVERWRITE=OVERWRITE)
> Source:
> /usr/local/itt/idl/lib/reverse.pro
> +----+
```

Since IDLWAVE does this as well (in context with C-c ?), but also includes system routines (even when not available as .pro files), I thought I'd mention how it does so. Since around IDL 6.2, ITT has

distributed with IDL an XML file which describes all of their routines, system variables, object classes, etc.: `idl/help/idl_catalog.xml`. It also gives detailed information on calling syntax, keywords, etc. Here's some simple code to parse this XML file to print the calling syntax of all the system routines for the currently running version of IDL:

```
dom=obj_new('IDLffXMLDOMDocument',VALIDATION_MODE=0, $
    FILENAME=filepath('idl_catalog.xml',SUBDIR='help'))
catalog=((dom->GetFirstChild())->GetNextSibling())->GetNextSibling()
routines=catalog->GetElementsByTagName('ROUTINE')

for i=0,routines->GetLength()-1 do begin
    routine=routines->Item(i)
    name=routine->GetAttribute('name')
    syns=routine->GetElementsByTagName('SYNTAX')
    print,name,':'
    for j=0,syns->GetLength()-1 do $
        print,' Syntax: ',(syns->Item(j))->GetAttribute('name')
endfor

obj_destroy,dom
```

A better version would gather the keywords as well (as IDLWAVE does) and place them in the syntax. Then perhaps tuck away a pair of string arrays into a common variable: sorted routine name, concatenated calling syntax. This could be used to look-up syntax (either as first priority, or second).

Beware of ALIAS_TO entries, which point to aliases for the actual ROUTINE block, and could obviously be worked around. You could also consider grabbing the CLASS elements, which contain METHOD children with their own syntax, etc. Might be very useful to have an interface to this on the command line.

JD

P.S. Is there a reason `a->b()->c` doesn't work in IDL? I'd have thought ``->'` was left to right associative?
