Subject: Re: convert string into the name of a variable
Posted by Paul van Delst on Wed, 05 Dec 2001 22:47:37 GMT
View Forum Message <> Reply to Message

Frederic Raison wrote:
>
> Hi!
>
> Is there anybody who could tell me if there is an IDL instruction, like
> in Matlab, able to convert a string into the name of a variable and
> vice-versa? I mean if I have defined :
>
> a=1
> b=2
> list=['a','b']
>
> I would like to write:
>
> print,somme_instruction(list[0])
>
> and to get:
>
> 1
>
> Thank you
>
> Frederic

this'll get yer started

```
IDL> a=149
IDL> b=273
IDL> list=['a','b']
IDL> result=execute('print,'+list[0])
     149
IDL> result=execute('print,'+list[1])
     273
```

if the execute works, result = 1.

--
Paul van Delst Religious and cultural
CIMSS @ NOAA/NCEP purity is a fundamentalist
Ph: (301)763-8000 x7274 fantasy
Fax:(301)763-8545 V.S.Naipaul

Subject: Re: convert string into the name of a variable
Posted by thompson on Wed, 05 Dec 2001 23:31:22 GMT
View Forum Message <> Reply to Message

Frederic Raison <frederic_raison@hotmai.com> writes:

> Hi!

> Is there anybody who could tell me if there is an IDL instruction, like
> in Matlab, able to convert a string into the name of a variable and
> vice-versa? I mean if I have defined :

> a=1
> b=2
> list=['a','b']

> I would like to write:

> print,somme_instruction(list[0])

> and to get:

> 1

> Thank you

> Frederic

The easiest way I can think of to do this would be

test=execute('print,'+list[0])

William Thompson

Subject: Re: convert string into the name of a variable
Posted by R.Bauer on Thu, 06 Dec 2001 14:05:54 GMT
View Forum Message <> Reply to Message

Frederic Raison wrote:
>
> Hi!
>
> Is there anybody who could tell me if there is an IDL instruction, like
> in Matlab, able to convert a string into the name of a variable and
> vice-versa? I mean if I have defined :
>

> a=1
> b=2
> list=['a','b']
>
> I would like to write:
>
> print,somme_instruction(list[0])
>
> and to get:
>
> 1
>
> Thank you
>
> Frederic

Dear Frederic,

as the two others wrote this could be done by execute.
But I like more to learn what's the idea to do this.
For what did you need this?

Another useful routine may be this one:
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_source/idl_
html/dbase/download/get_main_vars.tar.gz

or if you have a look in import_ascii you can see how it is done without
an execute

At the idl prompt type (This also will work in routines. But then you
have to use the level
keyword)

a=1
b=2
list=['a','b']

result=get_main_vars()
tn=strlowcase(tag_names(result))

;e.g.
ix=(where(tn eq list[0]))[0]

void = ROUTINE_NAMES(tn[ix], STORE=ROUTINE_NAMES(/LEVEL)-1, result.(ix))

I believe this would work in runtime but I am not sure.

The function execute will never work in runtime.

regards
Reimar

--
Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-1)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
http://www.fz-juelich.de/icg/icg1/
========== ======
a IDL library at ForschungsZentrum Juelich
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml

http://www.fz-juelich.de/zb/text/publikation/juel3786.html
========== ======

read something about linux / windows
http://www.suse.de/de/news/hotnews/MS.html

Subject: Re: convert string into the name of a variable
Posted by Pavel A. Romashkin on Thu, 06 Dec 2001 16:41:01 GMT
View Forum Message <> Reply to Message

Frederic Raison wrote:

> Is there anybody who could tell me if there is an IDL instruction, like
> in Matlab, able to convert a string into the name of a variable and
> vice-versa?

Runtime variable name definition. The history repeats itself for sure.
I was told by RSI developers when I started in IDL that once you find
yourself wanting this, it means that your progamming needs serious
phylosophy readjustment.
One side effect is that any time you save and restore a runtime-named
variable you need to again find out what the name of it is. And it is
not the only flaw.
I came to IDL from IGOR Pro, where meaningful names are the cornerstone
of it all and programming holds together by explicit naming conventions.
But in IGOR it is so because you have all your variables at the global
(*main*) level. IDL allows a lot more flexible approach because it is a
reuseable development environment.

Cheers,

Pavel