Subject: IsA questions
Posted by Matthew Argall on Tue, 10 Mar 2015 00:03:48 GMT
View Forum Message <> Reply to Message

Why are an invalid objects and pointers not scalars?

```
IDL> print, size(1)
           0           2           1
IDL> print, size(ptr_new())
           0          10           1
IDL> print, size(obj_new())
           0          11           1
IDL> print, isa(1, /SCALAR)
  1
IDL> print, isa(ptr_new(), /SCALAR)
  0
IDL> print, isa(obj_new(), /SCALAR)
  0
```

How is the /STRING keyword different from setting Typename to "String"?

```
IsA('theString', 'String')
IsA('theString', /STRING)
```

Finally, is this confusing to anyone else? Why not a /DECIMAL keyword instead?

```
a = 1.0d
PRINT, ISA(a, /FLOAT)   ; 1
PRINT, ISA(a, 'Float')  ; 0
```

Subject: Re: IsA questions
Posted by Matthew Argall on Tue, 10 Mar 2015 00:42:40 GMT
View Forum Message <> Reply to Message

I answered part of my own question... /STRING is to identify IDL_String classes. I guess the /INTEGER and /NUMBER keywords, then, would refer to the IDL_Integer and IDL_Number classes. And /FLOAT?

e.g. http://exelisvis.com/docs/IDL_String.html

Subject: Re: IsA questions
Posted by chris_torrence@NOSPAM on Tue, 17 Mar 2015 16:59:45 GMT
View Forum Message <> Reply to Message

On Monday, March 9, 2015 at 6:42:43 PM UTC-6, Matthew Argall wrote:
> I answered part of my own question... /STRING is to identify IDL_String classes. I guess the /INTEGER and /NUMBER keywords, then, would refer to the IDL_Integer and IDL_Number classes. And /FLOAT?
>
> e.g. http://exelisvis.com/docs/IDL_String.html

Hi Matthew,

I admit that there are some inconsistencies in the ISA results. For objects and pointers the idea was that ISA would return 0 for null object or pointers. However, I forgot about the /SCALAR keyword. I've fixed this for IDL 8.5, so ISA (with no keywords) will still return 0 for null heap variables, but will return 1 with /SCALAR.

The Typename argument refers to the actual IDL type (FLOAT or DOUBLE), as opposed to the FLOAT keyword, which can accept either float or double. Similarly for the COMPLEX keyword. DECIMAL might have been another choice, but would have been confusing for complex types...

I have also beefed up the Typename argument so it now accepts the "IDL_Variable" classes such as IDL_Integer, IDL_Number, and IDL_Variable.

Cheers,
Chris

Subject: Re: IsA questions
Posted by Matthew Argall on Tue, 17 Mar 2015 19:16:11 GMT
View Forum Message <> Reply to Message

Thanks for the response! Am I right in assuming the behavior of /STRING is different from the "String" typename? I have IDL 8.2 so cannot actually test this. The documentation is not clear, either.

Also, IDL assumes that any number that is not complex is purely real, so a REAL keyword could be used in combination with a DECIMAL keyword.

Subject: Re: IsA questions
Posted by chris_torrence@NOSPAM on Tue, 17 Mar 2015 22:22:12 GMT
View Forum Message <> Reply to Message

On Tuesday, March 17, 2015 at 1:16:13 PM UTC-6, Matthew Argall wrote:
> Thanks for the response! Am I right in assuming the behavior of /STRING is different from the "String" typename? I have IDL 8.2 so cannot actually test this. The documentation is not clear, either.
>
> Also, IDL assumes that any number that is not complex is purely real, so a REAL keyword

could be used in combination with a DECIMAL keyword.

/STRING and "String" should be identical. I got tired of typing the quotes (and it seemed wasteful) so I added the keyword.

I'll think about REAL and DECIMAL, but at this point, I'll probably just let the dust settle for a while.

Thanks for the suggestions!
-Chris
