Subject: LIST "bug": .Remove on an empty list
Posted by Matt Haffner on Mon, 13 Dec 2010 19:53:32 GMT
View Forum Message <> Reply to Message

Although this is easy to code around by checking the length before
calling .Remove, I was surprised this just didn't return silently:

```
IDL> l=list(1, length=100)
IDL> help,l
L               LIST  <ID=1424588  NELEMENTS=100>
IDL> l.remove,/all
IDL> help,l
L               LIST  <ID=1424588  NELEMENTS=0>
IDL> l.remove,/all
% PTR_FREE: Pointer type required in this context: P.
% Error occurred at: LIST::REMOVE
%                   LIST::REMOVE
%                   $MAIN$
% Execution halted at: $MAIN$
```

Passing it along to help the diagnosing of cryptic errors ;)

mh

Subject: Re: LIST "bug": .Remove on an empty list
Posted by Michael Galloy on Wed, 15 Dec 2010 17:22:12 GMT
View Forum Message <> Reply to Message

On 12/15/10 9:15 AM, Adam Lefkoff wrote:
> On 12/14/2010 3:34 PM, Michael Galloy wrote:
>> On 12/14/10 9:39 AM, David Fanning wrote:
>>> P.S. Does anyone beside me and Dick Jackson see the
>>> danger in calling a new graphics system "new"? Coyote
>>> suggests the name Newest New Old Graphics System for
>>> the work I've been doing lately, but I don't know... :-(
>>
>> Yes, that's why I thought "function graphics" would be a good name.
>> Maybe some
>> kind of numbered system? I think we would be up to new 4.0 graphics
>> now? With
>> "old" graphics being direct graphics, then object graphics, Live
>> Tools, iTools,
>> and "new graphics"? Am I missing some other graphics systems?
>>
>> Mike
>
> I seem to recall that there was a precursor to Live Tools named

> "Insight" (circa 94/95). And if you're counting *all* the graphic
> systems, shouldn't "Watsyn" make the cut ??

Ah, Insight, yes. I'm just counting graphics systems in IDL, so I don't
think Watsyn counts.

Mike
--
www.michaelgalloy.com
Research Mathematician
Tech-X Corporation

Subject: Re: LIST "bug": .Remove on an empty list
Posted by David Fanning on Fri, 17 Dec 2010 17:37:51 GMT
View Forum Message <> Reply to Message

Paulo Penteado writes:

> And I find "if (~l) then ..." much more convenient than "if
> (l.isempty()) then ...".

IsEmpty has the value, of course, of letting you know what
you were thinking months ago when you wrote the darn program. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: LIST "bug": .Remove on an empty list
Posted by penteado on Fri, 17 Dec 2010 18:03:51 GMT
View Forum Message <> Reply to Message

On Dec 17, 3:37 pm, David Fanning <n...@dfanning.com> wrote:
> Paulo Penteado writes:
>> And I find "if (~l) then ..." much more convenient than "if
>> (l.isempty()) then ...".
>

> IsEmpty has the value, of course, of letting you know what
> you were thinking months ago when you wrote the darn program. :-)

I would still say the same for the test of the object's truth value.
Other modern languages have used this for a while, and it is a common
convention that empty containers are false if empty, true otherwise.

Subject: Re: LIST "bug": .Remove on an empty list
Posted by Paul Van Delst[1] on Fri, 17 Dec 2010 18:12:41 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
> Paulo Penteado writes:
>
>> And I find "if (~l) then ..." much more convenient than "if
>> (l.isempty()) then ...".
>
> IsEmpty has the value, of course, of letting you know what
> you were thinking months ago when you wrote the darn program. :-)

I agree.

```
if (l.isempty()) then
```

is self-documenting to the poor souls that follow who are tasked with maintaining/extending the code (and I include the
orig author in that group! :o).

But

```
if (~l) then
```

not so much. To quote Bob Martin from "Clean Code":

"The problem isn't the simplicity of the code but the /implicity/ of the code: the degree to which the context is not
explicit in the code itself"

I'm not advocating that code should be understandable to the point where your grandma can figure out what you're trying
to do (been there, done that, not good), but new hires with limited experience (i.e. not computer science/programmer
types) should be able to easily grok what's going on since they will pretty much not know that "lists currently already
overload their truth value". :o)

cheers,