Subject: Re: Bug or desired behavior in lambda functions?
Posted by PMallas on Tue, 09 Jun 2015 13:09:19 GMT
View Forum Message <> Reply to Message

On Monday, June 8, 2015 at 1:34:12 PM UTC-4, Paulo Penteado wrote:
> Hello,
>
> Despite what the documentation might suggest, and one might expect, it seems there is no way
to use string variables to make lambda functions.
>
> If I try to make a lambda function with the code inside lambda(), all is well:
>
> IDL> l=lambda('x: x^2')
> IDL> l(2)
>   4
>
> However, if I put the same code in a string and pass it to lambda, it does not accept the code:
>
> IDL> expr='x: x^2'
> IDL> l=lambda(expr)
> % LAMBDA: Code must be of the form "arg1,arg2,... : statement"
>
> I expect this is caused by IDL parser's special behavior when it encounters lambda(), which is
what allows this
>
> IDL> l=lambda(x: x^2)
>
> So I think when I put a variable name inside the lambda() call, the parser thinks is a string
literal, and tries to parse the variable name as code.
>
> Is this intended behavior, or a bug?
>
> Paulo

I experimented with the same thing, I found this to work:

IDL> expr='x: x^2
IDL> l=lambda(expr)
% LAMBDA: Code must be of the form "arg1,arg2,... : statement"
% Execution halted at: $MAIN$
IDL> l=lambda(''+expr)
IDL> l(2)
        4