

Ruby - Bug #13337

Eval and Later Defined Local Variables

03/20/2017 06:36 PM - faraz (Faraz Yashar)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	2.4.0	
		Backport: 2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN
Description		
I've noticed the following behavior with eval:		
<pre>def example eval("foo") # => nil, but would expect a NameError eval("bar") # => NameError foo = 1 end</pre>		
I'm assuming this behavior is expected, since "eval" executes in the context of the existing YARV instructions and local table, but I'm not certain since I haven't seen this behavior documented anywhere.		

Associated revisions

Revision 0bd5f846df2f6268f653773e0cd4a20e2a944646 - 06/25/2019 04:52 PM - jeremyevans (Jeremy Evans)

Document local variable interactions with eval

Fixes [Bug #13337]

Revision 0bd5f846df2f6268f653773e0cd4a20e2a944646 - 06/25/2019 04:52 PM - jeremyevans (Jeremy Evans)

Document local variable interactions with eval

Fixes [Bug #13337]

Revision 0bd5f846 - 06/25/2019 04:52 PM - jeremyevans (Jeremy Evans)

Document local variable interactions with eval

Fixes [Bug #13337]

History

#1 - 05/19/2017 06:52 AM - ko1 (Koichi Sasada)

This is specification.

- (1) parse example method and define foo.
- (2) In eval method, compile given string ("foo") and defined foo was used (without any problem).

So it is documentation issue.
Can anyone write about it?

#2 - 06/24/2019 09:05 PM - jeremyevans0 (Jeremy Evans)

- File doc-eval-local-var.patch added

ko1 (Koichi Sasada) wrote:

This is specification.

- (1) parse example method and define foo.
- (2) In eval method, compile given string ("foo") and defined foo was used (without any problem).

So it is documentation issue.
Can anyone write about it?

Attached is a documentation patch. However, a committer with more knowledge of the internals should probably check to see if it is technically accurate.

#3 - 06/24/2019 09:11 PM - chrissseaton (Chris Seaton)

This is as specified here

https://github.com/ruby/spec/blob/db74bf3313b4d49650417ba91b4e2d67cc253dc5/core/kernel/eval_spec.rb#L31-L68

#4 - 06/25/2019 07:20 AM - nobu (Nobuyoshi Nakada)

jeremyevans0 (Jeremy Evans) wrote:

Attached is a documentation patch.

Seems fine.

A question:

However, local variables set inside the call to `+eval+` will not be reflected in the surrounding scope.

Isn't "set" vague a little, between "defined" and "assigned"?

#5 - 06/25/2019 04:53 PM - jeremyevans (Jeremy Evans)

- Status changed from Open to Closed

Applied in changeset [git|0bd5f846df2f6268f653773e0cd4a20e2a944646](https://github.com/ruby/spec/commit/0bd5f846df2f6268f653773e0cd4a20e2a944646).

Document local variable interactions with eval

Fixes [Bug [#13337](#)]

#6 - 06/25/2019 05:43 PM - nobu (Nobuyoshi Nakada)

"local variables assigned inside the call to `+eval+`" sounds like a just assignment to a local variable defined outside, doesn't it?

#7 - 06/25/2019 05:58 PM - jeremyevans0 (Jeremy Evans)

nobu (Nobuyoshi Nakada) wrote:

"local variables assigned inside the call to `+eval+`" sounds like a just assignment to a local variable defined outside, doesn't it?

It could definitely be interpreted that way. I'll reword it to make it more clear. Thanks!

Files

doc-eval-local-var.patch	1.63 KB	06/24/2019	jeremyevans0 (Jeremy Evans)
--------------------------	---------	------------	-----------------------------