

Ruby - Feature #18913

Add object name to the NoMethodError error message: undefined method `method' for `class' in `object_name'

07/14/2022 08:04 AM - stillhart (Fabian Stillhart)

Status:	Rejected	
Priority:	Normal	
Assignee:		
Target version:		

Description

Hi

I think the NoMethodError error message is missing the object name. The object name is already present in NameError. Currently one needs to dig into the source code and debug it in order to find which variable or method is actually affected.

The current error message looks like this:

```
bar = nil
bar.i_wish_i_saw_the_name_bar
# (irb):00:in `<main>': undefined method `i_wish_i_saw_the_name_bar' for nil:NilClass (NoMethodError)
```

I wish that it was more like this

```
bar = nil
bar.i_wish_i_saw_the_name_bar
# (irb):00:in `<main>': undefined method `i_wish_i_saw_the_name_bar' for nil:NilClass in `bar' (NoMethodError)
#                                     ^^^ It should mention the object name
```

The NameError error message supports this already, and I think it avoids a whole lot of issue tickets, stackoverflow searches and more. See NoMethodError vs NameError

<https://trends.google.ch/trends/explore?date=today%205-y&q=ruby%20%20NoMethodError.ruby%20NameError>

```
foo
# (irb):00:in `<main>': undefined local variable or method `foo' for main:Object (NameError)
#                                     ^^^ the object name is mentioned
```

One can always use the backtrace to find the affected line, but even that won't help in the following situation:

```
a = OpenStruct.new
b = nil
c = nil
who_failed = a.foo & b.foo & c.foo
# (irb):00:in `<main>': undefined method `foo' for nil:NilClass (NoMethodError)
```

I would add the object name at the end of the exception message not to break compatibility. I hope most users use regex to match the exception message.

I don't know how to program c but I think the source is located here

<https://github.com/ruby/ruby/blob/554befb/proc.c#L1959>

I assume there will be some edge cases around meta programming and procs. Anyway, I think we should show them if available. I think we all spent many hours tracking down when we have a type/method mismatch, and this would have helped greatly.

History

- #1 - 07/14/2022 08:04 AM - stillhart (Fabian Stillhart)
- Tracker changed from Bug to Feature
- ruby -v deleted (ruby 3.1.2p20 (2022-04-12 revision 4491bb740a) [x86_64-linux])

- Backport deleted (2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN)

#2 - 07/14/2022 08:31 AM - mame (Yusuke Endoh)

I don't understand what you mean by "object name", but did you try Ruby 3.1? It shows a code snippet that caused NameError and NoMethodError.

```
$ ruby test.rb
test.rb:5:in `<main>': undefined method `foo' for nil:NilClass (NoMethodError)
```

```
who_failed = a.foo & b.foo & c.foo
              ^^^^
```

```
$ ruby -v
ruby 3.1.2p20 (2022-04-12 revision 4491bb740a) [x86_64-linux]
```

#3 - 07/20/2022 02:48 AM - mame (Yusuke Endoh)

- Status changed from Open to Rejected

I close this ticket. If you have any opinion, please let me know.