

## CommonRuby - Feature #15458

### Automatic Exception#cause print in IRB

12/24/2018 07:07 AM - jnchito (Junichi Ito)

<b>Status:</b>	Open
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
According to Ruby 2.6 NEWS, the following feature will be added:	
Print cause of the exception if the exception is not caught and printed its backtraces and error message [Feature <a href="#">#8257</a> ]	
I expected it would be applied in IRB too, but it wasn't. Here is my example:	
<pre>&gt; begin &gt; 1/0 &gt; rescue =&gt; e &gt; e.message &gt; end Traceback (most recent call last):   5: from /Users/jnchito/.rbenv/versions/2.6.0-rc1/bin/irb:23:in `<main>'   4: from /Users/jnchito/.rbenv/versions/2.6.0-rc1/bin/irb:23:in `load'   3: from /Users/jnchito/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/irb-0.9.6/exe/irb: 11:in `<top (required)="">'   2: from (irb):1   1: from (irb):4:in `rescue in irb_binding' NoMethodError (undefined method `message' for #&lt;ZeroDivisionError: divided by 0&gt;) Did you mean?  message</top></main></pre>	
On the other hand, when I run that code via ruby command, cause of the exception is automatically printed:	
<pre>\$ ruby sample.rb Traceback (most recent call last):   1: from sample.rb:2:in `<main>' sample.rb:2:in `/': divided by 0 (ZeroDivisionError)   1: from sample.rb:1:in `<main>' sample.rb:4:in `rescue in &lt;main&gt;': undefined method `message' for #&lt;ZeroDivisionError: divided by 0 &gt; (NoMethodError) Did you mean?  message</main></main></pre>	
I feel it would be more natural and useful if the new feature would work in IRB too.	