

Ruby master - Feature #9400

Respect constant lookup when using `raise`

01/12/2014 05:03 AM - wardrop (Tom Wardrop)

Status:	Open
Priority:	Normal
Assignee:	
Target version:	
Description	
<p>When raising an error without defining an exception class, <code>::RuntimeError</code> appears to be a hard-coded default. I propose that this be changed so proper constant lookup rules are respected when looking up <code>RuntimeError</code>. For example, I would expect both of the <code>raise</code> calls in this example to be identical and use the redefined <code>RuntimeError</code> class with the <code>hello</code> method:</p>	
<pre>module Test class RuntimeError < RuntimeError def hello puts 'Hi there!' end end begin raise RuntimeError rescue => e e.hello #=> Hi there! end begin raise rescue => e e.hello #=> undefined method `hello' for RuntimeError:RuntimeError end end</pre>	
<p>I would find this behavior to be of quite some convenience. A lot (most?) libraries only have a single error class, and this would allow developers to implicitly raise exceptions of the appropriate class.</p>	
<p>Thoughts?</p>	

History

#1 - 07/12/2019 03:36 AM - jeremyevans0 (Jeremy Evans)

- Backport deleted (1.9.3: UNKNOWN, 2.0.0: UNKNOWN, 2.1: UNKNOWN)

- ruby -v deleted (2.1.0)

- Tracker changed from Bug to Feature