

Ruby - Bug #9376

Passing do/end block inside decorated method fails, while passing {} block works.

01/07/2014 12:21 PM - jemc (Joe McIlvain)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	2.1.0-p0	Backport:	1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONTNEED
Description =begin Since method (((def))) returns the symbol in 2.1, it opens the door for a decorator pattern in which the method def is the argument to a method. This currently works, unless the method happens to call another method and pass it a ((do/end))) block. However, using ((({}))) instead causes no syntax error. Here is the code to reproduce: def foo(meth); end x = Object.new foo def bar(arg) x.tap { } #=> (no problem here) x.tap do end #=> syntax error, unexpected keyword_do_block, expecting keyword_end (SyntaxError) end =end			
Related issues: Is duplicate of Ruby - Bug #9308: def's return value causes errors			
		Closed	12/27/2013

History

- #1 - 01/07/2014 01:00 PM - nobu (Nobuyoshi Nakada)
 - Description updated
 - Status changed from Open to Closed
 - Backport changed from 1.9.3: UNKNOWN, 2.0.0: UNKNOWN, 2.1: UNKNOWN to 1.9.3: DONTNEED, 2.0.0: REQUIRED, 2.1: REQUIRED
- #2 - 02/18/2014 11:27 PM - naruse (Yui NARUSE)
 - Backport changed from 1.9.3: DONTNEED, 2.0.0: REQUIRED, 2.1: REQUIRED to 1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONTNEED