

Ruby - Bug #5349

Add back documentation for Proc#call and Proc#[]

09/22/2011 05:20 AM - akahn (Alexander Kahn)

Status:	Closed	
Priority:	Normal	
Assignee:	drbrain (Eric Hodel)	
Target version:	1.9.3	
ruby -v:	1.9.3	
Description		Backport:
When matz added support for Proc#=== (https://github.com/akahn/ruby/commit/dd27b8f4577a96385a29d91e4bd75fb547213fc9), rdoc stopped outputting the documentation for Proc#call and Proc#[], which are far more commonly used. This patch adds back the documentation for these more commonly used methods, and while adding a description of Proc#===.		

Associated revisions

Revision b7ea625b - 10/03/2011 10:59 PM - drbrain (Eric Hodel)

- proc.c (proc_call): Fix documentation of Proc#call vs Proc#===.
[Ruby 1.9 - Bug #5349]
- git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@33390 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision e1960cdb - 10/08/2011 09:42 AM - kosaki (Motohiro KOSAKI)

merge revision(s) 33390:

```
* proc.c (proc_call): Fix documentation of Proc#call vs Proc#===.  
[Ruby 1.9 - Bug #5349]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@33430 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 10/04/2011 07:58 AM - drbrain (Eric Hodel)

I decided to keep #call and #=== separate, so I moved the === documentation above the call documentation and added a Document-method line.

#2 - 10/04/2011 07:59 AM - drbrain (Eric Hodel)

- Status changed from Open to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r33390.
Alexander, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- proc.c (proc_call): Fix documentation of Proc#call vs Proc#===.
[Ruby 1.9 - Bug [#5349](#)]

Files

0001-Add-back-documentation-for-Proc-call-etc.patch	1.67 KB	09/22/2011	akahn (Alexander Kahn)
---	---------	------------	------------------------