

Ruby trunk - Feature #4475

default variable name for parameter

03/06/2011 10:48 PM - jordi (jordi polo)

Status: Assigned	
Priority: Normal	
Assignee: matz (Yukihiko Matsumoto)	
Target version:	
Description	
=begin	
There is a very common pattern in Ruby:	
object.method do variable_name	
variable_name doing something	
end	
Many times in fact the name of the object is so self explanatory that we don't care about the name of the variable of the block. It is common to see things like :	
@my_sons.each { s s.sell_to_someone }	
or	
Account.all.each { a my_account << a.money }	
People tend to choose s or a because we have the class or the object name just there to remind you about the context.	
I would like to know if can be a good idea to have a default name for that parameter. I think it is Groovy that does something like:	
Account.all.each { my_account << it.money }	
Where it is automagically filled and it doesn't need to be declared.	
I think it is as readable or more (for newbies who don't know what is) and we save some typing :)	
=end	

History

#1 - 03/21/2011 06:57 PM - wardrop (Tom Wardrop)

=begin

I like the suggestion. The magic variable I'd use for this pattern would be 'this'. For example:

```
posts.each { this.author = 'Santa Clause' }
```

The rule would be: The first argument of any block would be accessible from within the block through the special variable 'this'.

=end

#2 - 03/18/2012 07:05 PM - nahi (Hiroshi Nakamura)

- Description updated

- Category set to core

- Status changed from Open to Assigned

- Assignee set to matz (Yukihiko Matsumoto)

#3 - 11/20/2012 09:26 PM - mame (Yusuke Endoh)

- Target version set to 2.6

#4 - 12/25/2017 06:14 PM - naruse (Yui NARUSE)

- Target version deleted (2.6)

#5 - 04/19/2018 07:51 AM - knu (Akinori MUSHA)

I would like the feature, but we have many things to think about.

We would not be able to make "it" a reserved keyword because that would destroy all existing RSpec code written in tens of thousands of projects.

If "it" were to be implemented as a method, how could we make it work inside of a BasicObject instance? What if a method of the same name was defined?

If "it" were to be implemented as a local variable, should its name be included in local_variables?

All things considered, I guess the variable name would have to be \$-something, if any.

#6 - 04/19/2018 11:42 AM - matz (Yukihiro Matsumoto)

This is a very interesting idea but at the same time, it's difficult to keep compatibility. At least simple addition of it does not work well.

Matz.