

Ruby master - Bug #1487

String#each_char must return self

05/19/2009 02:27 PM - marcandre (Marc-Andre Lafortune)

Status: Closed	
Priority: Normal	
Assignee:	
Target version: 1.9.2	
ruby -v: ruby 1.9.2dev (2009-05-19 trunk 23489) [i386-darwin9.7.0]	Backport:
Description =begin str.each_char{} currently returns a copy of str. It should return self. This also affects Ruby 1.8.7 Poor man's diff for rb_str_each_char: <pre>rb_encoding *enc;</pre> <ul style="list-style-type: none">• VALUE orig = str; RETURN_ENUMERATOR(str, 0, 0); str = rb_str_new4(str); ...• return str;• return orig; =end	

History

#1 - 05/23/2009 10:21 AM - matz (Yukihiko Matsumoto)

=begin
Hi,

In message "Re: [ruby-core:23499] [Bug #1487] String#each_char must return self" on Tue, 19 May 2009 14:27:54 +0900, Marc-Andre Lafortune redmine@ruby-lang.org writes:

|str.each_char{} currently returns a copy of str. It should return self. This also affects Ruby 1.8.7

Before accepting the proposal, could you explain why we should return the original string? We haven't defined the return value of each_char yet.

matz.

=end

#2 - 05/23/2009 11:38 PM - manveru (Michael Fellinger)

=begin
On Sat, 23 May 2009 10:20:50 +0900
Yukihiko Matsumoto matz@ruby-lang.org wrote:

Hi,

In message "Re: [ruby-core:23499] [Bug #1487] String#each_char must return self" on Tue, 19 May 2009 14:27:54 +0900, Marc-Andre Lafortune redmine@ruby-lang.org writes:

|str.each_char{} currently returns a copy of str. It should return self. This also affects Ruby 1.8.7

Before accepting the proposal, could you explain why we should return the original string? We haven't defined the return value of each_char yet.

matz.

string = 'foo'

string.object_id == string.each_codepoint{|x|.object_id

```
# => true
string.object_id == string.each_byte{|x|.object_id
# => true
string.object_id == string.each_line{|x|.object_id
# => true
string.object_id == string.each_char{|x|.object_id
# => false
```

I was hoping for a little bit of consistency as well.

```
--
^ manveru

=end
```

#3 - 05/24/2009 12:01 AM - rue (Eero Saynatkari)

```
=begin
Excerpts from Michael Fellingner's message of Sat May 23 17:38:35 +0300 2009:
```

On Sat, 23 May 2009 10:20:50 +0900
Yukihiko Matsumoto matz@ruby-lang.org wrote:

Before accepting the proposal, could you explain why we should return the original string? We haven't defined the return value of each_char yet.

```
matz.
```

```
string = 'foo'

string.object_id == string.each_codepoint{|x|.object_id
```

=> true

```
string.object_id == string.each_byte{|x|.object_id
```

=> true

```
string.object_id == string.each_line{|x|.object_id
```

=> true

```
string.object_id == string.each_char{|x|.object_id
```

=> false

I was hoping for a little bit of consistency as well.

Also true for Array#each and so on. Whichever way it goes, they should all be the same.

Philosophically, returning self from a non-mutating iterator is the most straightforward thing to do and makes it possible to avoid the cost of the dup.

```
--
Magic is insufficiently advanced technology.

=end
```

#4 - 05/24/2009 12:34 AM - matz (Yukihiko Matsumoto)

```
=begin
Hi,
```

In message "Re: [ruby-core:23526] Re: [Bug #1487] String#each_char must return self" on Sun, 24 May 2009 00:00:52 +0900, Eero Saynatkari ruby-ml@kittensoft.org writes:

|> I was hoping for a little bit of consistency as well.

|
|Also true for Array#each and so on. Whichever way it goes,
|they should all be the same.
|
|Philosophically, returning self from a non-mutating iterator
|is the most straightforward thing to do and makes it possible
|to avoid the cost of the dup.

The string may be modified in the block, so that you cannot avoid the cost of the dup in this case. OK, consistency is the reason? What else?

```
matz.
```

=end

#5 - 05/24/2009 12:47 AM - rue (Eero Saynatkari)

=begin
Excerpts from Yukihiro Matsumoto's message of Sat May 23 18:34:35 +0300 2009:

Hi,

In message "Re: [ruby-core:23526] Re: [Bug #1487] String#each_char must return self"
on Sun, 24 May 2009 00:00:52 +0900, Eero Saynatkari ruby-ml@kittensoft.org writes:

|> I was hoping for a little bit of consistency as well.
|
|Also true for Array#each and so on. Whichever way it goes,
|they should all be the same.
|
|Philosophically, returning self from a non-mutating iterator
|is the most straightforward thing to do and makes it possible
|to avoid the cost of the dup.

The string may be modified in the block, so that you cannot avoid the cost of the dup in this case. OK, consistency is the reason? What else?

I suppose this is true, although I am in the "enough String to hang themselves" camp on the matter if someone thinks it a good idea to muck with the object being iterated on (and not using a mutating iterator.) The user can always do the dup themselves. Might it warrant a mention in the method documentation?

--
Magic is insufficiently advanced technology.

=end

#6 - 05/24/2009 01:07 AM - matz (Yukihiro Matsumoto)

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

=begin
Applied in changeset r23552.
=end

#7 - 05/24/2009 01:22 AM - matz (Yukihiro Matsumoto)

=begin
Hi,

In message "Re: [ruby-core:23528] Re: [Bug #1487] String#each_char must return self"
on Sun, 24 May 2009 00:47:19 +0900, Eero Saynatkari ruby-ml@kittensoft.org writes:

|I suppose this is true, although I am in the "enough String
|to hang themselves" camp on the matter if someone thinks it
|a good idea to muck with the object being iterated on (and
|not using a mutating iterator.) The user can always do the
|dup themselves.

I'm not sure if I understand what you mean. Do you mean "since the users always dup themselves, internal dup is not needed"? If so, I disagree, especially this case. Strings with Multibyte characters are so easily hindered by modification, and it is not always easy to make sure strings are not modified in the block. So if only single object allocation can save me (copy-on-write is used internally, so no contents are copied), I'd love to pay the cost.

```
matz.
```

```
=end
```