## 2.2 String#unicode_normalize method does not appear in generated API docs

**04/20/2015 02:39 PM - jrochkind (jonathan rochkind)**

### Details
- **Status:** Closed
- **Priority:** Normal
- **Assignee:** duerst (Martin Dürst)
- **Target version:**
  - `ruby -v:` ruby 2.2.1p85 (2015-02-26 revision 49769) [x86_64-darwin12.0]
  - Backport: 2.0.0: UNKNOWN, 2.1: UNKNOWN, 2.2: UNKNOWN

### Description
The new in 2.2 String#unicode_normalize and String#unicode_normalized? methods are very useful.

But they do not seem to appear in the generated ruby API docs, which means many people won't realize they exist.

http://ruby-doc.org/core-2.2.0/String.html

Does not mention any unicode_normalize method. Should it?

There is generated docs for a UnicodeNormalize module at [http://ruby-doc.org/stdlib-2.2.0/libdoc/unicode_normalize/rdoc/UnicodeNormalize.html](http://ruby-doc.org/stdlib-2.2.0/libdoc/unicode_normalize/rdoc/UnicodeNormalize.html), but it documents methods with different names, just normalize and normalized?. I think these are the same methods. I am not sure if this module is what is used in String? I am not sure why the method names differ, or why they don't show up directly in the generated String API docs.

If they could show up in the generated String API docs, with the correct method names, it would make it easier for developers to discover their existence and proper method names and functionality.

### Associated revisions
- **Revision 58329 - 04/12/2017 06:07 PM - stomar (Marcus Stollsteimer)**
  - nodoc UnicodeNormalize module
  - lib/unicode_normalize/normalize.rb: [DOC] nodoc the internal UnicodeNormalize module.
  - lib/unicode_normalize/tables.rb: ditto.
  - template/unicode_norm_gen.tmpl: ditto.

### History
- **#1 - 04/24/2015 09:26 AM - duerst (Martin Dürst)**
  -Assignee set to drbrain (Eric Hodel)

I have assigned this bug to Eric in order to get information on how to proceed from him; assigning it back to me with some instructions on what to do is fine.

jonathan rochkind wrote:

The new in 2.2 String#unicode_normalize and String#unicode_normalized? methods are very useful.

Thanks!

But they do not seem to appear in the generated ruby API docs, which means many people won't realize they exist.

http://ruby-doc.org/core-2.2.0/String.html

Does not mention any unicode_normalize method. Should it?

I guess yes.

There is generated docs for a UnicodeNormalize module at [http://ruby-doc.org/stdlib-2.2.0/libdoc/unicode_normalize/rdoc/UnicodeNormalize.html](http://ruby-doc.org/stdlib-2.2.0/libdoc/unicode_normalize/rdoc/UnicodeNormalize.html), but it documents methods with different names, just normalize and normalized?. I think these are the same methods. I am not sure if this module is what is used in String? I am not sure why the method names differ, or why they don't show up directly in the generated String API docs.
The methods in the UnicodeNormalize module are the backend implementations for the methods on String. Except for people interested in how the methods on String are implemented, it doesn't make much sense for these (or anything else from UnicodeNormalize) to appear in the public documentation. If somebody knows the magic trick for excluding this module from the documentation, please tell me. [It won't hurt too much if it stays in the documentation, though.]

If they could show up in the generated String API docs, with the correct method names, it would make it easier for developers to discover their existence and proper method names and functionality.

The actual documentation for String#unicode_normalize, String#unicode_normalize!, and String#unicode_normalized? is in lib/unicode_normalize.rb (see e.g. http://svn.ruby-lang.org/cgi-bin/viewvc.cgi/trunk/lib/unicode_normalize.rb?revision=49064&view=markup). http://ruby-doc.org/core-2.2.0/String.html shows that the information is taken from various files (complex.c, pack.c, rational.c, string.c, transcode.c), but this seems to miss lib/unicode_normalize.rb.

I'm not sure why this happens, or what can be done about it, at this time or in the long term. I have therefore assigned this issue to Eric Hodel and hope to hear back from him with some advice as to how to proceed.

#2 - 04/12/2017 06:37 PM - stomar (Marcus Stollsteimer)
duerst (Martin Dürst) wrote:

[...] it doesn't make much sense for these (or anything else from UnicodeNormalize) to appear in the public documentation. If somebody knows the magic trick for excluding this module from the documentation, please tell me.

The magic trick is :nodoc:. I removed the documentation with r58329.

Regarding the docs for String#unicode_normalize:

The docs do appear in the generated documentation. However, ruby-doc.org separates core and stdlib documentation, so http://ruby-doc.org/core-2.4.1/String.html has all the methods defined in string.c, rational.c, ..., while the methods defined in lib/unicode_normalize.rb appear under http://ruby-doc.org/stdlib-2.4.1/libdoc/unicode_normalize/rdoc/String.html.

On the other hand, docs.ruby-lang.org shows all methods under the corresponding class's documentation, see https://docs.ruby-lang.org/en/trunk/String.html#method-i-unicode_normalize.

Also, ri String lists those methods at the very end of the class documentation.

Hope this helps.

#3 - 04/17/2017 11:07 AM - duerst (Martin Dürst)

stomar (Marcus Stollsteimer) wrote:

The docs do appear in the generated documentation. However, ruby-doc.org separates core and stdlib documentation, so http://ruby-doc.org/core-2.4.1/String.html has all the methods defined in string.c, rational.c, ..., while the methods defined in lib/unicode_normalize.rb appear under http://ruby-doc.org/stdlib-2.4.1/libdoc/unicode_normalize/rdoc/String.html.

Also, ri String lists those methods at the very end of the class documentation.

Would it make sense to convert the code in lib/unicode_normalize.rb to C and move it to string.c? That would also eliminate enc/prelude.rb.

#4 - 04/28/2017 01:45 PM - shyouhei (Shyouhei Urabe)

- Status changed from Open to Assigned

#5 - 05/04/2017 02:29 AM - duerst (Martin Dürst)

- Assignee changed from drbrain (Eric Hodel) to duerst (Martin Dürst)

#6 - 05/04/2017 02:31 AM - duerst (Martin Dürst)

- Status changed from Assigned to Closed

Fixed in r58555 and related revisions, by converting Ruby code in lib/unicode_normalize.rb to C code in string.c.

#7 - 05/05/2017 07:48 AM - Hannmac (Hans Mackowiak)
duerst (Martin Dürst) (Martin Dürst):

i see your changes with unicode_normalize_common, i only asking because i think that rb_scan_args would be better for that, because:
1. it does check for the args better
2. also it might be easier to use rb_funcallv instead of two calls with rb_funcall

#8 - 05/09/2017 11:21 AM - duerst (Martin Dürst)

Hanmac (Hans Mackowiak) wrote:

i see your changes with unicode_normalize_common,
i only asking because i think that rb_scan_args would be better for that, because:

1. it does check for the args better
2. also it might be easier to use rb_funcallv instead of two calls with rb_funcall

Implemented with r58618, thanks!
(I thought this might be slightly slower, but I wasn't able to see any difference, so I went ahead.)