

Ruby - Feature #5352

How about using <> to represent Here Document?

09/22/2011 11:16 PM - yimutang (Joey Zhou)

Status:	Rejected	
Priority:	Normal	
Assignee:	matz (Yukihiro Matsumoto)	
Target version:	3.0	
Description <p>"<<" is a frequently used method in text manipulation. Meanwhile, it's also the beginning token of a here-document. Sometimes it may be confusing to newbies.</p> <p>Unlike Perl, in which "<>" is a very very frequently used operator to read lines from a filehandle, in Ruby, "<>" has no meaning. So I think it can be used to represent Here Document, so as to reduce the possibility of misunderstanding method "<<" and here-document token "<<". And In my opinion, <> is more clear than <<, because it looks like kind of brackets.</p> <p>For example,</p> <pre>str = <SECT1>.upcase + <SECT2>.downcase aaaaaa SECT1 XXXXXX SECT2</pre> <p>may be clearer than:</p> <pre>str = <<SECT1.upcase + <<SECT2.downcase</pre>		

History

#1 - 09/23/2011 10:37 AM - duerst (Martin Dürst)

If anything is done in this direction, I'd prefer

```
str = <>SECT1.upcase + <>SECT2.downcase
```

over

```
str = <SECT1>.upcase + <SECT2>.downcase
```

It seems to stick out much better.

#2 - 09/23/2011 10:53 AM - yeban (Anurag Priyam)

On Fri, Sep 23, 2011 at 7:08 AM, Martin Dürst duerst@it.aoyama.ac.jp wrote:
[...]

If anything is done in this direction, I'd prefer
Ã str

#3 - 09/24/2011 05:25 AM - nobu (Nobuyoshi Nakada)

Joey Zhou wrote:

Unlike Perl, in which "<>" is a very very frequently used operator to read lines from a filehandle, in Ruby, "<>" has no meaning. So I think it can be used to represent Here Document, so as to reduce the possibility of misunderstanding method "<<" and here-document token "<<". And In my opinion, <> is more clear than <<, because it looks like kind of brackets.

I'm somewhat negative.
It looks like Perl's "<>" but is pretty different, so I'm afraid that it would be also confusing.

#4 - 09/26/2011 12:06 AM - yimutang (Joey Zhou)

Nobuyoshi Nakada wrote:

I'm somewhat negative.

It looks like Perl's "<>" but is pretty different, so I'm afraid that it would be also confusing.

There are a few things different in Ruby and Perl.

\$foo is not a scalar, @bar is not an array, %w is not a hash

"re = /pattern/" in Ruby means assigning a regexp to re, but in Perl "\$re = /pattern/" means "\$re = \$_ =~ /pattern/"...

So I think the difference is not a problem, some differences already exist.

"<<" of Perl has two meanings: shifting bits and here document. One is for the integers and the other is for the string, maybe leading to less confusing. However, in Ruby, "<<" is instance method of String and IO/File, and here document is also string, the probability of confusing is larger, I'm afraid.

#5 - 03/27/2012 03:07 AM - mame (Yusuke Endoh)

- Status changed from Open to Assigned

- Assignee set to matz (Yukihiro Matsumoto)

#6 - 11/20/2012 09:50 PM - mame (Yusuke Endoh)

- Target version set to 3.0

#7 - 04/19/2018 07:49 AM - naruse (Yui NARUSE)

- Description updated

#8 - 04/19/2018 07:52 AM - matz (Yukihiro Matsumoto)

- Status changed from Assigned to Rejected

Rejected. I don't want to change the basic syntax (without major benefit). Besides that, I want to keep <> for future syntax extension.

Matz.