

Ruby - Bug #11116

The spec of String#dump

05/03/2015 10:36 AM - mame (Yusuke Endoh)

Status:	Closed	
Priority:	Normal	
Assignee:	mame (Yusuke Endoh)	
Target version:		
ruby -v:	ruby 2.2.1p85 (2015-02-26 revision 49769) [x86_64-linux]	Backport: 2.0.0: UNKNOWN, 2.1: UNKNOWN, 2.2: UNKNOWN
Description The current spec says: <pre>call-seq: str.dump -> new_str</pre> Produces a version of +str+ with all non-printing characters replaced by <code><code>\nnn</code></code> notation and all special characters escaped. <pre>"hello \n ' '".dump #=> "\"hello \\n '\\\""</pre> \nnn must be \xnn now. In addition, I've expected String#dump to return a string that evaluates to an original string (except singleton methods, object id, etc.) when evaled. Is this a right expectation? If so, it would be good to officially include the mention in the spec. What do you think? -- Yusuke Endoh mame@ruby-lang.org		

History

#1 - 01/21/2019 08:52 AM - mame (Yusuke Endoh)

- Status changed from Open to Closed

Committed at r66894. Closing

#2 - 01/21/2019 10:54 AM - Eregon (Benoit Daloze)

Does that preserve the encoding of the String though?

What about String#inspect, does it also eval() to itself?

#3 - 01/21/2019 01:20 PM - mame (Yusuke Endoh)

Eregon (Benoit Daloze) wrote:

Does that preserve the encoding of the String though?

The short answer: yes.

If the encoding is ASCII-compatible, it is preserved via its encoding of the resulting string.

```
s = "Hello 000000".encode("Windows-31J")
s = s.dump
puts s          #=> "Hello \x82\xB1\x82\xF1\x82\xC9\x82\xBF\x82\xCD"
p s.encoding    #=> #<Encoding:Windows-31J>
s = eval(s)
p s.encoding    #=> #<Encoding:Windows-31J>
puts s.encode("UTF-8") #=> Hello 000000
```

If the encoding is not ASCII-compatible, the dumped string has an explicit force_encoding.

```
s = "Hello 000000".encode("UTF-16LE")
s = s.dump
```

```
puts s          #=> "H\x00e\x00l\x00l\x00o\x00 \x00S0\x930k0a0o0".dup.force_encoding("UTF-16LE")
s = eval(s)
puts s.encode("UTF-8") #=> Hello       
```

I guess that there might be other subtle edge cases that I don't know. It is difficult for me to write the detailed document.

What about `String#inspect`, does it also `eval()` to itself?

No, it is not guaranteed. `String#inspect` is just for human, so there is no annoying hack like the above non-ASCII-compatible encoding. (I have encountered another subtle case, but I cannot remember...)