Ruby trunk - Feature #14919

Add String#byteinsert

07/18/2018 05:18 AM - aycabta (aycabta .)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		

Description

It's important for multibyte String editing. Unicode grapheme characters sometimes have plural code points. In text editing, software sometimes should add a new code point to an existing grapheme character. String#byteinsert is important for it.

I implemented by pure Ruby in my code.
https://github.com/aycabta/reline/blob/b17e5fd61092adfd7e87d576301e4e19a4d9e6d8/lib/reline/line_editor.rb#L255-L260

History

#1 - 07/18/2018 05:18 AM - aycabta (aycabta .)

- Backport deleted (2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN)

- Tracker changed from Bug to Feature

#2 - 07/18/2018 06:41 AM - duerst (Martin Dürst)

aycabta (aycabta .) wrote:

> It's important for multibyte String editing. Unicode grapheme characters sometimes have plural code points. In text editing, software sometimes should add a new code point to an existing grapheme character. String#byteinsert is important for it.

Can you explain this a bit more? Editing of code points is easily possible with String#[]=; there is no need to use byteinsert.

#3 - 07/18/2018 07:25 AM - aycabta (aycabta .)

duerst (Martin Dürst) wrote:

> Editing of code points is easily possible with String#[]=; there is no need to use byteinsert.

Input from CLI

In CLI tool, all characters come as each of the bytes. All multibyte characters are split. In the middle of a line, a software should use an insertion of a new character but not a replacement.

Yank

In the middle of a line, yank manipulation needs #byteinsert for multibyte editing.

#4 - 07/18/2018 09:20 AM - duerst (Martin Dürst)

aycabta (aycabta .) wrote:

> duerst (Martin Dürst) wrote:
>
> > Editing of code points is easily possible with String#[]=; there is no need to use byteinsert.
>
> ## Input from CLI
>
> In CLI tool, all characters come as each of the bytes. All multibyte characters are split.

On the lowest level, characters indeed come in as a string of bytes. But it would be wrong to insert individual bytes into a string unless these bytes are also characters. It would just lead to mojibake.

The right thing to do is to collect a (small) number of bytes, check how many bytes are needed to form one or more characters, insert these characters into the string, and keep the remaining bytes for further processing (wait until more bytes arrive so that we get more complete

codepoints/characters).

> In the middle of a line, a software should use an insertion of a new character but not a replacement.

Insertion of characters can be done with String#[]=.

Yank

> In the middle of a line, yank manipulation needs #byteinsert for multibyte editing.

I still don't see why. You don't want to insert bytes, you want to insert characters, so that the String is correctly encoded at all times.

#5 - 07/18/2018 01:29 PM - shevegen (Robert A. Heiler)

I don't have a specific opinion on the suggestion itself; Martin raised some valid points, in my opinion. But I wanted to comment on something else.

There have been some suggestions to the developer meeting, as recently as 8 hours ago; so probably just shortly before the developer meeting started:

https://bugs.ruby-lang.org/issues/14861

This is a very short time frame. I would like to suggest to give a little bit more time before the developer meeting, so that other people can also comment on the suggestions. Something like +24 hours or so if it has not yet discussed; I feel that ~8 hours without any real possibility for a discussion is very, very short.