Ruby trunk - Feature #13241

Method(s) to access Unicode properties for characters/strings

02/22/2017 08:02 AM - duerst (Martin Dürst)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		

Description

[This is currently an exploratory proposal.]

Onigmo allows Unicode properties in regular expressions. With this, it's e.g. possible to check whether a string contains some Hiragana:

```
"ABC □ DEF" =~ /\p{hiragana}/
```

However, it is currently impossible to ask for e.g. the script of a character. I propose to add a method (or some methods) to String to be able to get such properties. Various (to some extent conflicting) examples:

```
"A□□".script => :latin # returns script of first character only
```

```
"A□□".script => [:latin, :hiragana, :katakana] # returns array of property values
```

```
"A□□".property(:script) => :latin # returns specified property of first character only
```

```
"A□□".property(:script) => [:latin, :hiragana, :katakana] # returns array of specified properties'
 values
```

```
"A□□".properties([:script, :general_category]) => [[:latin, :Lu], [:hiragana, :Lo], [:katakana, :L
o]]
                        # returns arrays of property values, one array per character
```

The interface is still in flux, comments welcome!

Implementation depends on #13240.

In Python, such functionality (however, quite limited in property coverage, and not directly on String) is available in the standard library (see https://docs.python.org/3/library/unicodedata.html).

Related issues:

Related to Ruby trunk - Feature #13240: Change Unicode property implementatio...	**Open**
Related to Ruby trunk - Feature #14618: Add display width method to String fo...	**Open**

History

#1 - 02/22/2017 08:04 AM - duerst (Martin Dürst)

- Related to Feature #13240: Change Unicode property implementation in Onigmo from inversion lists to direct lookup added

#2 - 02/22/2017 09:06 AM - matz (Yukihiro Matsumoto)

I am neutral about the proposal, but the method names are too generic. It should be prefixed by unicode_ for example.

Matz.

#3 - 02/22/2017 09:17 AM - rbjl (Jan Lelis)

Great idea, I'd love to have such capabilities built into the language!

I've recently build this for scripts, blocks, and general categories on Ruby level (see https://github.com/janlelis/unicode-scripts), so let me share some thoughts on the API:

- I think, it should be always *plural methods* which return a list of properties used in the string, since Ruby does not distinguish between single characters and strings. The first example would then rather be: "A□□".scripts => [:hiragana, :katakana, :latin] (like the fourth example). I find it better that it would always return an array than being confused by the fact that it would only consider the first character.
- With the same reasoning, I would go for having only a properties method, and no singular property method

- Although I kind of like the .properties([:script, :general_category]) API, it can be a little confusing when using the proposed *plural methods* approach: It implicitly switches its mode of operation to character by character, soley based on the passed argument being an array. I'd suggest to make this explicit, maybe by using another method such as .each_properties, just going with each_char.properties (probably cannot get optimized properly), or using a keyword argument like by_char: true
- Should there be only a .properties method (which could be used with scripts, blocks, general categories, etc.) or should there also be individual methods (like .scripts, .blocks, …)? I think both ways would be acceptable, but I like the idea of having individual methods for the most important properties.
- A little more bikeshedding: Maybe the properties should be returned as strings instead of symbols. They represent some kind of data, so to me it feels like strings are the more appropriate choice. Another example, if we have such functionality for blocks as well, "Miscellaneous Mathematical Symbols-B" would have to returned as a symbol - which just does not look so good. This is only about the values returned, all method arguments would still be symbols/keyword arguments.

What do you all think?

#4 - 02/22/2017 09:22 AM - rbjl (Jan Lelis)

I think prefixing such methods with unicode_ would be no problem. While it's a little verbose, it still reads good:

- "bla".unicode_scripts
- "blubb".unicode_properties(:general_categories)

and so on. Also it is consistent with the unicode_normalize API.

#5 - 02/22/2017 05:21 PM - shevegen (Robert A. Heiler)

Jan Lelis wrote:

> I think, it should be always plural methods which return a list of properties used in the
> string, since Ruby does not distinguish between single characters and strings. The first
> example would then rather be: "A□□".scripts => :hiragana, :katakana, :latin.

I agree in the sense that your example given makes more sense than the first example,
where:

"A□□".script => :latin # returns script of first character only

Only returned one result. I understand it was just an example, but it confused me because
I wondered what happened to the other characters?

I like the name "property" or "properties" more than "script" - script sounds a bit
non-descript (pun intended!).

Since matz said that it should be indicative of unicode, e. g. with a unicode_prefix,
the example by Jan Lelis would seem good:

"string here".unicode_properties(optional_args)

Other name suggestions:

.unciode_category
.unciode_categories
.unciode_tokenset
.unciode_token_set
.unciode_tokens

And similar perhaps.

PS: By the way, what should it return for an empty string like ""? Or numbers
or similar semi-common tokens?

#6 - 04/09/2018 07:28 AM - duerst (Martin Dürst)

- Related to Feature #14618: Add display width method to String for CLI added