

Ruby master - Feature #14783

String#chars_at / String#bytes_at

05/23/2018 10:23 AM - sos4nt (Stefan Schüßler)

| | |
|--|--------|
| Status: | Open |
| Priority: | Normal |
| Assignee: | |
| Target version: | |
| Description | |
| I just wanted to extract characters at certain indices from a string and noticed that there's no values_at counterpart for String. | |
| I'd therefore like to propose two new String methods: | |
| <ul style="list-style-type: none">• chars_at(selector, ...) → new_str• bytes_at(selector, ...) → new_str | |
| which work basically like Array#values_at , e.g.: | |
| <pre>string = 'hello, world!' string.chars_at(0, 5, 7, 12) #=> "h,w!" string.chars_at(0..4, 7..11) #=> "helloworld"</pre> | |

History

#1 - 05/23/2018 10:30 AM - Hanmac (Hans Mackowiak)

[sos4nt \(Stefan Schüßler\)](#)

why does it return a new string instead of array of strings?

you might like this:

```
string = 'hello, world!'
[0, 5, 7, 12].map(&string.method(:[])).join #=> "h,w!"
[0..4, 7..11].map(&string.method(:[])).join #=> "helloworld"
```

#2 - 05/23/2018 10:35 AM - zverok (Victor Shepelev)

```
string = 'hello, world!'
string.chars.values_at(0, 5, 7, 12)
# => ["h", ",", "w", "!"]
string.chars.values_at(0, 5, 7, 12).join
# => "h,w!"
```



#3 - 05/23/2018 10:38 AM - sos4nt (Stefan Schüßler)

Hanmac (Hans Mackowiak) wrote:

why does it return a new string instead of array of strings?

Because String#[] also returns a string in such case:

```
a = [1, 2, 3, 4]
s = "1234"

a[1..2] #=> [2, 3]
s[1..2] #=> "23"
```

Accordingly:

```
a.values_at(1..2) #=> [2, 3]
s.chars_at(1..2) #=> "23"
```

#4 - 05/24/2018 05:04 AM - nobu (Nobuyoshi Nakada)

sos4nt (Stefan Schüßler) wrote:

Because `String#[]` also returns a string in such case:

It's a different, single argument case.

I don't think that `values_at` should return same class as the receiver, as `Hash#values_at`.

#5 - 05/24/2018 01:18 PM - shevegen (Robert A. Heiler)

Is the frequency of `#bytes_at` common? I understand the use case stated by Stefan (extract multiple indices via one method call on a `String`) but `#bytes_at` may seem to be quite rare.

As for returning a `String` or `Array`, I personally expect `#chars` on a `String` object to return an `Array`, so intuitively I would assume `#chars_at` to also return an `Array`.