

## Ruby master - Feature #10240

### `String#to\_a`

09/14/2014 02:51 PM - sawa (Tsuyoshi Sawada)

<b>Status:</b>	Rejected
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
I often want to exclude the empty strings from an array of strings. A use case is to join an array of strings with ", ", removing the empty ones, as follows:	
<pre>["Foo", "", "Bar"].reject(&amp;:empty?).join(", ") # =&gt; "Foo, Bar"</pre>	
In a similar situation with arrays of arrays, the empty ones can be excluded by using * and Array#to_a:	
<pre>[*[1, 2], *[], *[3, 4]] # =&gt; [1, 2, 3, 4]</pre>	
I would like to propose String#to_a defined as follows:	
<pre>class String   def to_a; empty? ? [] : [self] end end</pre>	
Then we can do:	
<pre>[*"Foo", *"", *"Bar"].join(", ")</pre>	

### History

#### #1 - 09/14/2014 07:58 PM - matz (Yukihiro Matsumoto)

I understand your intention, but I think proposed behavior is bit awkward for to\_a of String class. As you know to\_a stands for conversion to array, I don't think "" -> [] is natural conversion from strings.

Matz.

#### #2 - 09/16/2014 12:13 AM - sawa (Tsuyoshi Sawada)

Matz is right. I was wrong about the idea. In the cases I showed, I should have had nil values instead of empty strings. I will withdraw this proposal.

#### #3 - 01/04/2020 06:18 PM - sawa (Tsuyoshi Sawada)

- Description updated

- Subject changed from String#to\_a to `String#to\_a`

#### #4 - 01/04/2020 06:21 PM - sawa (Tsuyoshi Sawada)

I do not feel this proposal to be natural any more. Please close this issue.

#### #5 - 01/04/2020 08:30 PM - Eregon (Benoit Daloze)

- Status changed from Open to Rejected