

Ruby master - Feature #13777

Array#delete_all

07/30/2017 07:12 AM - k0kubun (Takashi Kokubun)

Status:	Rejected
Priority:	Normal
Assignee:	
Target version:	
Description	
I want Array#delete_if which returns array of deleted values.	
For following code,	
<pre>array = ["a", "aa", "ab", "bb", "c"] result = {} until array.empty? key = array.first group, array = array.partition { v v.start_with?(key) } result[key] = group end result #=> {"a"=>["a", "aa", "ab"], "bb"=>["bb"], "c"=>["c"]}</pre>	
With Array#delete_all, This would be able to be written in more elegant way like:	
<pre>array = ["a", "aa", "ab", "bb", "c"] result = {} until array.empty? key = array.first result[key] = array.delete_all { v v.start_with?(key) } end result #=> {"a"=>["a", "aa", "ab"], "bb"=>["bb"], "c"=>["c"]}</pre>	
This is simplified source code of real use case in Haml: https://github.com/haml/haml/blob/923a0d78874fe1d369f8c7a0bf77f67b2c2139bb/lib/haml/attribute_compiler.rb#L75-L76	
This grouping task is necessary for Haml optimization.	
Do you know simpler way to write this with existing methods?	

History

#1 - 07/30/2017 01:36 PM - shevegen (Robert A. Heiler)

I myself usually use .reject! and .select! and then apply the reverse prior to that if I need to keep these entries as well (or respectively without the '!').

My approach also needs more lines.

If I understood it correctly so then you want to have an operation where you can not only manipulate the variable at hand, but also additionally select the entries that were deleted. I do not know of a better way than the one you showed though, at the least not with any fewer lines of code.

#2 - 10/19/2017 07:28 AM - matz (Yukihiro Matsumoto)

- Status changed from Open to Rejected

The name delete_all is not acceptable. This method works as a modifying version of partition. The name does not indicate the fact. This can be achieved by

```
result = []
ary.delete_if do|e|
  if cond(e)
    result << e
    true
  end
end
```

end

If the use-case is frequent and needs to be supported by the core, we will re-investigate. Please reopen.

Matz.