

## Ruby trunk - Feature #14197

### `Enumerable#{select,reject}` accept a pattern argument

12/18/2017 12:42 PM - znz (Kazuhiro NISHIYAMA)

<b>Status:</b>	Open
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
<p>#all?, #any?, #none?, and #one? accept a pattern argument since 2.5.0. But #select, and #reject don't.</p> <p>The features are exist as #grep, and #grep_v, but there are hard to remember for me when I use #select, or #reject.</p> <p>So I want to write</p> <pre>collection.reject(/re/)</pre> <p>instead of</p> <pre>collection.reject { item  /re/ =~ item }</pre> <p>nor</p> <pre>collection.grep_v(/re/)</pre>	
<b>Related issues:</b>	
Related to Ruby trunk - Feature #11286: [PATCH] Add case equality arity to En... <span style="float: right;">Closed</span>	

## History

### #1 - 12/18/2017 02:12 PM - znz (Kazuhiro NISHIYAMA)

- Related to Feature #11286: [PATCH] Add case equality arity to Enumerable's sequence predicates. added

### #2 - 12/20/2017 09:29 PM - shevegen (Robert A. Heiler)

I think this suggestion looks ok, unless I may have missed something.

The main difference Kazuhiro is suggesting, appears to be the shorter notation via regex given to .reject() rather than use the (longer) block variant.

Here is the link to #grep:

<https://ruby-doc.org/core-2.1.0/Enumerable.html#method-i-grep>

which shows one example:

```
c = IO.constants
c.grep(/SEEK/)      #=> [:SEEK_SET, :SEEK_CUR, :SEEK_END]
```

On a side note, while I have used .grep(), I have never used .grep\_v() and it is a bit mysterious to me what grep\_v does from the name alone. :)

### #3 - 12/25/2017 06:15 PM - naruse (Yui NARUSE)

- Target version deleted (2.6)

### #4 - 01/11/2018 09:32 PM - inopinatus (Joshua GOODALL)

I have personally found it useful to implement Regexp#to\_proc as in <https://bugs.ruby-lang.org/issues/7883> which permits collection.reject(&/re/) and more besides.

### #5 - 03/21/2018 03:49 PM - xavriley (Xavier Riley)

This issue is also related <https://bugs.ruby-lang.org/issues/9602>

One interesting point there is the case where a pattern and a block are given:

```
%w{ant bear cat}.select(/bear/) {|x| x === "cat" }
```

This issue also applies to the implementation from the linked issue for #all?(pattern)but this is already implemented

```
>> %w{ant bear cat}.all?(String) {|x| x === Integer }  
=> true
```

#### **#6 - 03/21/2018 04:10 PM - matz (Yukihiro Matsumoto)**

The point is that the phrase "hard to remember" in the OP is bit weak when we already have methods with the proposed behavior (grep and grep\_v).

Matz.

#### **#7 - 06/19/2018 10:56 AM - byroot (Jean Boussier)**

The point is that the phrase "hard to remember" in the OP is bit weak when we already have methods with the proposed behavior (grep and grep\_v).

Indeed. However I think there is a case to be made for consistency.

If this works:

```
%(foo bar).all?(String)
```

I do expect this to work as well.

```
%(foo bar).reject(String)
```

It is true that then select would end up being an alias of grep and reject an alias of grep\_v, but I actually see it as a good thing.