Ruby - Feature #11049

Enumerable#grep_v (inversed grep)

04/08/2015 10:23 AM - sorah (Sorah Fukumori)

Status:	Closed
Priority:	Normal
Assignee:	matz (Yukihiro Matsumoto)
Target version:	

Description

sometime I want to do grep -v like operation:

```
%w(aaa bbb ccc).reject { |x| /b/ === x } #=> ["aaa", "ccc"]
```

We already have Enumerable#grep, so I propose to add Enumerable#grep_v.

```
%w(aaa bbb ccc).grep(/b/) #=> ["bbb"]
%w(aaa bbb ccc).grep_v(/b/) #=> ["aaa", "ccc"]
```

Naming / Interface

This idea is mentioned at DevelopersMeeting20150408Japan by me. Matz has said "I don't disagree for the feature. So this remains naming (interface) problem."

I'm not sure grep_v is the best name for this feature; feedback are welcomed.

Ideas

- grep_v(pattern) (the first patch)
- grep(pattern, inverse: true)
- grep!(pattern)

Related issues:

Related to Ruby - Feature #5588: add negation flag (v) to Regexp	**Closed**
Related to Ruby - Feature #9602: Logic with `Enumerable#grep`	**Feedback**
Related to Ruby - Feature #8921: Allow select, reject, etc to accept a regex	**Open**

History

#1 - 04/08/2015 11:18 AM - recursive-madman (Recursive Madman)

How about implementing Regexp#invert instead?

```
  /b/.invert #=> /[^(?:b)]/
```

Like this for example:

```
class Regexp
  def invert
    self.class.new("[^(?:#{source})]")
  end
end

%w(aaa bbb ccc).grep(/b/) #=> ["bbb"]
%w(aaa bbb ccc).grep(/b/.invert) #=> ["aaa", "ccc"]
```

#2 - 04/08/2015 11:21 AM - Eregon (Benoit Daloze)

Recursive Madman wrote:

> How about implementing Regexp#invert instead?
>
> Like this for example:
>
> ```
> class Regexp
>   def invert
> ```

```
    self.class.new("[^(?:#{source})]")
  end
end
```

That only works for some regexps, I doubt there is a proper and well defined inverted regexp for any Regexp.

#3 - 04/08/2015 11:22 AM - Eregon (Benoit Daloze)

grep(pattern, inverse: true) seems the least surprising to me.
But of course it does not buy much compared to #reject.

#4 - 04/08/2015 11:31 AM - recursive-madman (Recursive Madman)

Benoit Daloze wrote:

> That only works for some regexps, I doubt there is a proper and well defined inverted regexp for any Regexp.

Could you provide an example that doesn't work?

#5 - 04/08/2015 11:42 AM - akr (Akira Tanaka)

Recursive Madman wrote:

> Benoit Daloze wrote:
>
> > That only works for some regexps, I doubt there is a proper and well defined inverted regexp for any Regexp.
>
> Could you provide an example that doesn't work?

```
class Regexp
  def invert
    self.class.new("[^(?:#{source})]")
  end
end

%w(aaa bbb ccc).grep(/abc/) #=> []
%w(aaa bbb ccc).grep(/abc/.invert) #=> []
```

#6 - 04/08/2015 12:00 PM - nobu (Nobuyoshi Nakada)

invert looks same as my proposal, Regexp#!.

#7 - 04/08/2015 04:45 PM - sawa (Tsuyoshi Sawada)

How is this proposal different from #9602? I thought that Matz had already suggested to modify select, and Sam Rawlins had implemented select and reject.

#8 - 04/09/2015 05:44 AM - sorah (Sorah Fukumori)

- Assignee set to matz (Yukihiro Matsumoto)

I didn't know/hear that.

@matz (Yukihiro Matsumoto) what do you think for now?

#9 - 04/11/2015 05:56 AM - akr (Akira Tanaka)

- Related to Feature #5588: add negation flag (v) to Regexp added

#10 - 04/11/2015 12:50 PM - akr (Akira Tanaka)

- Related to Feature #9602: Logic with `Enumerable#grep` added

#11 - 04/19/2015 08:30 PM - haraldb (Harald Böttiger)

Recursive Madman wrote:

> How about implementing Regexp#invert instead?

As grep using the case statement this would be ignoring a lot of cases that would be handy:

array.invert_grep(Array) # Select all elements that are not arrays.

#12 - 05/14/2015 06:31 AM - matz (Yukihiro Matsumoto)

grep_v seems OK. Accepted.

Matz.

#13 - 05/14/2015 10:44 AM - sorah (Sorah Fukumori)

- Status changed from Open to Closed

Thanks, committed at r50491, r50492.

#14 - 03/03/2016 04:39 AM - shyouhei (Shyouhei Urabe)

- Related to Feature #8921: Allow select, reject, etc to accept a regex added

Files

grepv.patch	4.24 KB	04/08/2015	sorah (Sorah Fukumori)