Ruby master - Bug #18953

`Array#uniq` doesn't evaluate the given block when the size of the array is one

08/02/2022 02:15 AM - ttanimichi (Tsukuru Tanimichi)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>ruby 3.1.2p20 (2022-04-12 revision 4491bb740a) [arm64-darwin21]</td>
</tr>
<tr>
<td>Backport:</td>
<td>2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN</td>
</tr>
</tbody>
</table>

Description

Array#uniq doesn't evaluate the given block when the size of the array is one. Is this expected behavior?

$ ruby -e '[42, 43].uniq { _1.foo }'; puts true'
-e:1:in `block in <main>': undefined method `foo' for 42:Integer (NoMethodError)

$ ruby -e '[42, 43].uniq { _1.foo }'; puts true

```
42
```
Did you mean?  floor
from -e:1:in `uniq'
from -e:1:in `<main>'

$ ruby -e '[42].uniq { _1.foo }'; puts true

```
true
```

History

#1 - 08/03/2022 04:52 AM - mrkn (Kenta Murata)

I hope this behavior should be expected. However, sort_by, max_by, and min_by call the given blocks even if the receiver array has only one item.

```
irb(main):001:0> [42].sort_by { _1.foo }
(irb):1:in `block in <top (required)>': undefined method `foo' for 42:Integer (NoMethodError)
 Did you mean?  floor
from (irb):1:in 'each'
from (irb):1:in 'sort_by'
from (irb):1:in '<main>'
from /home/mrkn/.rbenv/versions/3.1.0/lib/ruby/gems/3.1.0/gems/irb-1.4.1/exe/irb:11:in `<top (required)>'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `load'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `<main>'
irb(main):002:0> [42].max_by { _1.foo }
(irb):2:in `block in <top (required)>': undefined method `foo' for 42:Integer (NoMethodError)
 Did you mean?  floor
from (irb):2:in 'each'
from (irb):2:in 'max_by'
from (irb):2:in '<main>'
from /home/mrkn/.rbenv/versions/3.1.0/lib/ruby/gems/3.1.0/gems/irb-1.4.1/exe/irb:11:in `<top (required)>'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `load'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `<main>'
irb(main):003:0> [42].min_by { _1.foo }
(irb):3:in `block in <top (required)>': undefined method `foo' for 42:Integer (NoMethodError)
 Did you mean?  floor
from (irb):3:in 'each'
from (irb):3:in 'min_by'
from (irb):3:in '<main>'
from /home/mrkn/.rbenv/versions/3.1.0/lib/ruby/gems/3.1.0/gems/irb-1.4.1/exe/irb:11:in `<top (required)>'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `load'
from /home/mrkn/.rbenv/versions/3.1.0/bin/irb:25:in `<main>''
```

#2 - 08/03/2022 06:12 AM - nobu (Nobuyoshi Nakada)

I think this is an expected behavior, but an implementation detail.

#3 - 08/09/2022 04:52 PM - jeremyevans0 (Jeremy Evans)

- Status changed from Open to Rejected