Ruby master - Feature #9071

Enumerable#slice_after

11/02/2013 01:21 AM - sawa (Tsuyoshi Sawada)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>akr (Akira Tanaka)</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
</tbody>
</table>

Description

I see in this discussion: [http://ruby.11.x6.nabble.com/ruby-dev-38392-Enumerable-gather-each-td3534746.html](http://ruby.11.x6.nabble.com/ruby-dev-38392-Enumerable-gather-each-td3534746.html) that Enumerable#slice_before was named as such, having in mind the possibility of Enumerable#slice_after being implemented in the future. I feel the former convenient, but believe the latter should be as well, and am wondering why the latter was not implemented at the same time. I request it to be implemented.

Associated revisions

Revision ddd15584 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Requested by Tsuyoshi Sawada. [ruby-core:58123] [Feature #9071]

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)
Requested by Tsuyoshi Sawada. [ruby-core:58123] [Feature #9071]

Revision 45981 - 05/18/2014 12:06 AM - akr (Akira Tanaka)

I implemented Enumerable#slice_after.

I found two request for this feature:
This issue and
http://stackoverflow.com/questions/22622156/how-to-implement-slice-after-or-group-certain-elements-with-certain-subsequent

I wrote the document with an example follows:

```ruby
% ./ruby -e 'lines = ["foo\n", "bar\\\n", "baz\n", "\n", "qux\n"]
```
This concatenates continuation lines.

More useful (catchy) examples may be helpful to persuade matz.

Any idea?

#3 - 05/12/2014 09:47 AM - akr (Akira Tanaka)
- File slice_after2.patch added

I updated the patch to simplify argument handling.

#4 - 05/17/2014 06:33 AM - matz (Yukihiro Matsumoto)
Accepted.

Matz

#5 - 05/18/2014 12:06 AM - akr (Akira Tanaka)
- Status changed from Open to Closed
- % Done changed from 0 to 100

Applied in changeset r45981.

- enum.c: Enumerable#slice_after implemented.
- enumerator.c: Enumerator::Lazy#slice_after implemented.

Requested by Tsuyoshi Sawada. [ruby-core:58123] [Feature #9071]

---

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>slice_after.patch</td>
<td>7.01 KB</td>
<td>05/10/2014</td>
<td>akr (Akira Tanaka)</td>
</tr>
<tr>
<td>slice_after2.patch</td>
<td>6.87 KB</td>
<td>05/12/2014</td>
<td>akr (Akira Tanaka)</td>
</tr>
</tbody>
</table>