Ruby - Feature #6847

Hash#extract

08/10/2012 02:01 AM - citizen428 (Michael Kohl)

Status:	Rejected	
Priority:	Normal	
Assignee:	matz (Yukihiro Matsumoto)	
Target version:	2.6	

Description

I often need a method to extract "subhashes" from hashes. I often monkey-patched this into hash and found "extract" to be a good name for the method. Afterwards I found out that there actually is a mutating method called "extract!" in ActiveSupport, so I guess the name is kinda established. Here's the Ruby version from my private extension library, if there's any interest in this I would try to re-implement it in C:

```
class Hash
def extract(*ks)
existing = keys & ks
Hash[existing.zip(values_at(*existing))]
end
end
```

History

#1 - 08/10/2012 02:53 AM - claytrump (Clay Trump)

Isn't that like the method slice in ActiveSupport?

On Thu, Aug 9, 2012 at 1:01 PM, citizen428 (Michael Kohl) <
citizen428@gmail.com> wrote:

> Issue #6847 has been reported by citizen428 (Michael Kohl).
>
> ---
>
> Feature #6847: Hash#extract
> https://bugs.ruby-lang.org/issues/6847
>
> Author: citizen428 (Michael Kohl)
> Status: Open
> Priority: Normal
> Assignee:
> Category: core
> Target version:
>
> I often need a method to extract "subhashes" from hashes. I often monkey-patched this into hash and found "extract" to be a good name for the method. Afterwards I found out that there actually is a mutating method called "extract!" in ActiveSupport, so I guess the name is kinda established. Here's the Ruby version from my private extension library, if there's any interest in this I would try to re-implement it in C:
>
> ```
> class Hash
> def extract(*ks)
> existing = keys & ks
> Hash[existing.zip(values_at(*existing))]
> end
> end
> ```
>
> --
> http://bugs.ruby-lang.org/

--

#2 - 08/11/2012 06:42 PM - citizen428 (Michael Kohl)

claytrump (Clay Trump) wrote:

> Isn't that like the method slice in ActiveSupport?

It's definitely like extract! in ActiveSupport, just non-destructive. I still think this is useful enough to be in core though. I needed it repeatedly, I know other people who regularly monkey-patch it into Hash and not everyone wants to drag ActiveSupport into their non-Rails projects. I don't insist on it of course, but it wouldn't be the first thing to make it from an extensions library into core, think of Symbol#to_proc for example.

#3 - 08/11/2012 11:55 PM - trans (Thomas Sawyer)

Ruby Facets has Hash#slice as well.

http://rdoc.info:8080/github/rubyworks/facets/master/Hash#slice-instance_method

+1 for its addition of Hash.

#4 - 08/13/2012 01:23 AM - claytrump (Clay Trump)

> It's definitely like extract! in ActiveSupport, just non-destructive.

There is no "extract!" in ActiveSupport, its "slice!" you mean.

So yeah, I've used that before, and would be nice in core. I think it
should be called 'slice' instead of 'extract'. It's a better name and it
already exists like that in ActiveSupport and in Facets...

--

#5 - 08/13/2012 06:52 AM - citizen428 (Michael Kohl)

claytrump (Clay Trump) wrote:

> There is no "extract!" in ActiveSupport, its "slice!" you mean.

http://apidock.com/rails/Hash/extract%21

File activesupport/lib/active_support/core_ext/hash/slice.rb, line 35

```
def extract!(*keys)
result = {}
keys.each {|key| result[key] = delete(key) }
result
end
```

#6 - 08/13/2012 08:06 AM - trans (Thomas Sawyer)

It important to understand the differences and to determine the best definition. ActiveSupport's #slice returns the same type of class, which I think is important, where as #extract! always returns a Hash. Also, #slice and #slice! are provided, where as there is only #extract! in ActiveSupport. Certainly we want both. ActiveSupport's #slice also has a special feature for #convert_key, which I think that would be nice for Ruby to have too, but it is not part of Ruby at this time. So that would obviously be omitted.

I concur with Clay that #slice is the better name b/c Array#[] and Array#slice are pretty much the same, and though it might be a little bit more different that general relationship should also hold for Hash.

#7 - 08/13/2012 12:53 PM - claytrump (Clay Trump)

Oh my, what a disaster. I wonder who made the crazy decision that hash
slice would returned a slice of the hash and slice! would return... the
receiver and not the slice of the hash. Especially since Array slice and
slice! return the same values...

So, sorry Michael, I was confused. Anyways, I think the most important is
slice and that's the correct name too, and it might be nice to have a
mutating version too. I'd name it slice! and too bad for ActiveRecord, but
that might not be the best way to go.

--

#8 - 11/20/2012 11:23 PM - mame (Yusuke Endoh)

- Status changed from Open to Assigned

- Assignee set to matz (Yukihiro Matsumoto)

- Target version set to 2.6

#9 - 11/20/2012 11:30 PM - matz (Yukihiro Matsumoto)

- Status changed from Assigned to Rejected

Use Hash#select which now returns an hash.

Matz.