

Ruby trunk - Feature #13563

Implement Hash#choice method.

05/14/2017 10:43 AM - babanba-n (matzbara masanao)

Status:	Closed
Priority:	Normal
Assignee:	
Target version:	
Description	
Hi,	
I propose Hash#choice method.	
It pick up key and value pairs in Hash like a below code.	
<pre>{ :a => 1, 2 => "2", "c" => true }.choice(:a, 2) # => { :a => 1, 2 => "2" } { :a => 1, 2 => "2", "c" => true }.choice("c", 10000) # => { "c" => true, 10000 => nil }</pre>	
This method is useful when Hash have many keys, but programer need few key's.	
For instance, it pick up personal data in ActiveRecord model's data, and migrate to new device.	
<pre>feature_phone_user = User.find(params[:src_user_id]) profile = feature_phone_user.attributes.choice(*%i[nickname email sex birthday prefecture]) smart_phone_user = User.new smart_phone_user.attributes = profile</pre>	
In other case, it pick up latest log from http request parameter.	
<pre>condition = params.to_h.choice("user_id", "service_id") latest_payment_log = PaymentLog.find_by(condition).last</pre>	
If this method exist, I guess that many cases change to comfortable just a little.	
Please consider about it.	
And I write it.	
https://github.com/baban/ruby/commit/4014ebe2ae261881669bef8369eb08aaecef2034	
Related issues:	
Related to Ruby trunk - Feature #8499: Importing Hash#slice, Hash#slice!, Has...	Closed

History

#1 - 05/14/2017 03:15 PM - babanba-n (matzbara masanao)

- Subject changed from Implement Hash#choice mtthod. to Implement Hash#choice method.

#2 - 05/14/2017 05:57 PM - shevegen (Robert A. Heiler)

I am neutral about the suggestion, neither pro or con.

I think the only thing that I want to point out is that .choice() is perhaps not completely clear. Perhaps .pick() would be better?

We have quite some methods already that allow us to .select() .reject(); and it also reminds me a bit about .shuffle() and .sample().

I guess this is not the fault of anyone by the way, we just do not have that many (one-word) english words that are for verbs/actions. So perhaps .choice() is a good name anyway.

#3 - 05/14/2017 08:05 PM - matthewd (Matthew Draper)

FWIW, ActiveSupport calls this Hash#slice -- <http://api.rubyonrails.org/classes/Hash.html#method-i-slice>

(With one small behavioural difference: keys not present in the source hash are also omitted from the result, so the second example would return { "c" => true }, not { "c" => true, 10000 => nil }.)

#4 - 05/16/2017 03:59 AM - babanba-n (matzbara masanao)

At first, I overlook Hash#slice method. Sorry...
And, I reconsider about it.

I feel that Hash#slice is wrong name and Hash#pick is better name.
If key is not exist, key omit or return nil.
It is difficult problem.

I think each way have usecase.

For example, below code is return first row.

```
Item = ItemStock.find_by({}) # return first row
```

For that reason, below code have possibility return unintended data.

```
item = ItemStock.find_by(params.to_h.slice("user_id", "item_id"))  
return render { result: "NG" } if item.amount.zero?  
item.consume  
render json: { result: "OK" }
```

So returning nil is more safety.

#5 - 05/25/2017 01:35 PM - saturnflyer (Jim Gay)

babanba-n (matzbara masanao) wrote:

I feel that Hash#slice is wrong name and Hash#pick is better name.
If key is not exist, key omit or return nil.
It is difficult problem.

I prefer Hash#pick over Hash#choice.

I would expect the missing keys to return whatever the default value of the hash is.

```
hash = Hash.new('foo')  
hash[:bar] = 'hello world'  
  
hash.pick(:baz, :bar) #=> {:baz => 'foo', :bar => 'hello world' }
```

#6 - 05/30/2017 09:56 AM - babanba-n (matzbara masanao)

Thanks reply, saturnflyer.
And I totally agree your propose.

Even if, this purpose is rejected.
I will fix code, and re-push it.

#7 - 06/10/2017 02:26 PM - babanba-n (matzbara masanao)

And, I fix to Hash#pick.

<https://github.com/baban/ruby/commit/fa9d1d6291e4507c5fdd997c269cd1cd68003899>

#8 - 06/10/2017 03:32 PM - Hanmac (Hans Mackowiak)

babanba-n (matzbara masanao) wrote:

And, I fix to Hash#pick.

<https://github.com/baban/ruby/commit/fa9d1d6291e4507c5fdd997c269cd1cd68003899>

in the test case, shouldn't the @h.default= "default" be before the pick?

#9 - 06/12/2017 02:24 PM - babanba-n (matzbara masanao)

Sorry, I'm so embarrassed.
<https://github.com/baban/ruby/commit/b3b7f5f701e9bfbbc946d27d3397adc1c99125f6>

#10 - 08/31/2017 06:12 AM - mrkn (Kenta Murata)

- Related to Feature #8499: Importing Hash#slice, Hash#slice!, Hash#except, and Hash#except! from ActiveSupport added

#11 - 09/01/2017 04:29 AM - matz (Yukihiro Matsumoto)

I vote for Hash#slice that is ActiveSupport compatible.

Matz.

#12 - 11/27/2017 02:01 PM - marcandre (Marc-Andre Lafortune)

- Status changed from Open to Closed