Ruby - Feature #11818

`Hash#compact`

12/15/2015 05:09 AM - sawa (Tsuyoshi Sawada)

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

Description

I request Hash#compact and Hash#compact! that remove the key-value pairs whose value is nil, as follows:

```
h1 = {a:, 1, b: nil, c: 2}
h1.compact # => {a: 1, c: 2}
h1 # => {a: 1, b: nil, c: 2}

h2 = {a:, 1, b: nil, c: 2}
h2.compact! # => {a: 1, c: 2}
h2 # => {a: 1, c: 2}

h3 = {a:, 1, c: 2}
h3.compact! # => nil
h3 # => {a: 1, c: 2}
```

I believe people have frequent need to do this.

Associated revisions

Revision bbd58fa2b956a33c035dbddc83e9efbe07ee7276 - 10/13/2016 08:06 AM - nobu (Nobuyoshi Nakada)

hash.c: add compact and compact! methods

- hash.c (rb_hash_compact, rb_hash_compact_bang): Removes nil
  values from the original hash, to port Active Support behavior.
  [Feature #11818]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@56414 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision bbd58fa2 - 10/13/2016 08:06 AM - nobu (Nobuyoshi Nakada)

hash.c: add compact and compact! methods

- hash.c (rb_hash_compact, rb_hash_compact_bang): Removes nil
  values from the original hash, to port Active Support behavior.
  [Feature #11818]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@56414 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision f087533b8ba13fa3f1765618067ab298b97ebea1 - 10/13/2016 08:15 AM - nobu (Nobuyoshi Nakada)

NEWS: [Feature #11818] [ci skip]

- NEWS: add Hash#compact and Hash#compact!. [Feature #11818]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@56415 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision f087533b - 10/13/2016 08:15 AM - nobu (Nobuyoshi Nakada)

NEWS: [Feature #11818] [ci skip]

- NEWS: add Hash#compact and Hash#compact!. [Feature #11818]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@56415 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

---

History

#1 - 12/15/2015 05:11 AM - sawa (Tsuyoshi Sawada)

Sorry, the code was invalid. It should be:

```
h1 = {a: 1, b: nil, c: 2}
h1.compact # => {a: 1, c: 2}
h1 # => {a: 1, b: nil, c: 2}

h2 = {a: 1, b: nil, c: 2}
h2.compact! # => {a: 1, c: 2}
h2 # => {a: 1, c: 2}

h3 = {a: 1, c: 2}
h3.compact! # => nil
h3 # => {a: 1, c: 2}
```

#2 - 12/15/2015 01:33 PM - danielpclark (Daniel P. Clark)

You can do this with Hash#delete_if

```
{a: 1, b: nil, c: 2}.delete_if {|k,v| v.nil?}
# => {:a=>1, :c=>2}
```

#3 - 12/15/2015 01:57 PM - sikachu (Prem Sichanugrist)

Active Support has this: http://api.rubyonrails.org/classes/Hash.html#method-i-compact

I'm +1 on porting this method. While I think it's possible using a block form and delete_if, this method has a good name and intention that could live by its own.

#4 - 01/03/2016 10:11 PM - dwfait (Dwain Faithfull)

Prem Sichanugrist wrote:

> Active Support has this: http://api.rubyonrails.org/classes/Hash.html#method-i-compact
>
> I'm +1 on porting this method. While I think it's possible using a block form and delete_if, this method has a good name and intention that could live by its own.

https://github.com/ruby/ruby/pull/1184/files

In anticipation of this being approved, I have raised a PR of a sample implementation, along with some benchmarks against the Active Support implementation and tests.

This is my first time contributing to Ruby, so I apologise if I've jumped the gun or done anything wrong - seemed like a good issue to pick up for someone new to the codebase, with no possible breaking changes, and some good performance gains to be had.

#5 - 07/22/2016 03:06 AM - mrkn (Kenta Murata)

- Status changed from Open to Assigned

- Assignee set to matz (Yukihiro Matsumoto)

#6 - 08/09/2016 07:06 AM - matz (Yukihiro Matsumoto)

Accepted.

Matz.

#7 - 10/13/2016 08:06 AM - nobu (Nobuyoshi Nakada)

- Status changed from Assigned to Closed

Applied in changeset r56414.

---

hash.c: add compact and compact! methods

- hash.c (rb_hash_compact, rb_hash_compact_bang): Removes nil
  values from the original hash, to port Active Support behavior.
  [Feature #11818]