

## Ruby - Feature #10072

### [PATCH] Implement Vector.basis

07/19/2014 07:39 AM - gogotanaka (Kazuki Tanaka)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	marcandre (Marc-Andre Lafortune)	
<b>Target version:</b>	2.2.0	
<b>Description</b> Standard basis vectors are really important in linear algebra.  And we usually need this when we use matrix or vector. (base conversion, principal component analysis...  This is why I implemented.  But I recognize Ruby should be not for mathematician but rubyist.  Ruby doesn't need too academic method.  So If you think it is too academic you can ignore my patches.  gogo.		

#### History

##### #1 - 07/19/2014 04:01 PM - gogotanaka (Kazuki Tanaka)

We can build standard euclidean space easily with this method.

```
[Vector.basis(3, 0), Vector.basis(3, 1), Vector.basis(3, 2)]  
=> [Vector[1, 0, 0], Vector[0, 1, 0], Vector[1, 0, 1]]
```

This patch is needed in <https://bugs.ruby-lang.org/issues/10074>

gogo.

##### #2 - 07/20/2014 07:10 PM - hsbt (Hiroshi SHIBATA)

- Category set to lib

- Status changed from Open to Assigned

- Assignee set to marcandre (Marc-Andre Lafortune)

- Target version set to 2.2.0

##### #3 - 10/02/2014 08:51 PM - marcandre (Marc-Andre Lafortune)

Looks good, but I'm wondering if we should use named arguments as it's really difficult to remember the order without looking it up.

```
Vector.basis(3, 0)  
# or  
Vector.basis(size: 3, index: 0)
```

Not sure.

##### #4 - 10/03/2014 09:19 PM - gogotanaka (Kazuki Tanaka)

@Marc-Andre Lafortune

I get your point. I admire your taste.

After some agonizing, I come up with implementing Vector.new (like Array.new) also.

And implement Vector#basis as for index.

So I mean, how about using Vector.new(3).basis(0) instead of Vector.basis(3, 0) ?

Just idea.

#### #5 - 10/04/2014 02:24 AM - marcandre (Marc-Andre Lafortune)

Interesting idea, but it has issues, like private new currently requiring an array, and it being two steps, the first one not too clear.

I think it's best to pick between the two possibilities I gave.

#### #6 - 10/05/2014 06:39 AM - duerst (Martin Dürst)

On 2014/10/03 05:51, [ruby-core@marc-andre.ca](mailto:ruby-core@marc-andre.ca) wrote:

Issue [#10072](#) has been updated by Marc-Andre Lafortune.

Looks good, but I'm wondering if we should use named arguments as it's really difficult to remember the order without looking it up.

```
Vector.basis(3, 0)
# or
Vector.basis(size: 3, index: 0)
```

Not sure.

Why not allow both? Ruby is flexible.

Regards, Martin.

#### #7 - 10/07/2014 07:59 PM - marcandre (Marc-Andre Lafortune)

- Status changed from Assigned to Closed

Martin Dürst wrote:

Why not allow both? Ruby is flexible.

Indeed.

In this case though, it would be a bit of an overkill I think.

Let's use named arguments. If basis becomes really popular and we want to make a shortcut we still can.

#### #8 - 10/07/2014 10:57 PM - gogotanaka (Kazuki Tanaka)

@Martin Dürst @Marc-Andre Lafortune

I agree it. Thank you so much.

#### Files

---

implement_Vector.basis.patch	1.59 KB	07/19/2014	<a href="#">gogotanaka (Kazuki Tanaka)</a>
------------------------------	---------	------------	--