Ruby master - Bug #5966

->{} doesn't work in BasicObject

02/04/2012 02:25 PM - ktsj (Kazuki Tsujimoto)

<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>ko1 (Koichi Sasada)</td>
</tr>
<tr>
<td>Target version:</td>
<td>2.0.0</td>
</tr>
<tr>
<td>ruby -v:</td>
<td>ruby 2.0.0dev (2012-02-03 trunk 34424) [x86_64-linux]</td>
</tr>
</tbody>
</table>

**Description**

```ruby
$ ./ruby -ve 'BasicObject.new.instance_eval { ->{} }'
ruby 2.0.0dev (2012-02-03 trunk 34424) [x86_64-linux]
-e:1:in block in <main>: undefined method 'lambda' for #<BasicObject:0x007ffff847ef28> (NoMethodError)
from -e:1:in instance_eval'
from -e:1:in'
```

I'm not sure it is a bug or not, but I think lambda syntax should work everywhere.

=end

**Associated revisions**

Revision 6a8d0097 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.
**Revision 34819 - 02/26/2012 02:26 AM - nobu (Nobuyoshi Nakada)**

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

**Revision 331d6b34 - 02/26/2012 03:26 AM - nobu (Nobuyoshi Nakada)**

merge revision(s) 34819:

```
* compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
* insns.def (send): no special deal for FCALL. self should be put on TOS instead.
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@34820 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**History**

**#1 - 02/11/2012 06:56 AM - nobu (Nobuyoshi Nakada)**
- File 0001-Bug-5966-lambda-in-BasicObject.patch added
- Status changed from Open to Assigned
- Assignee set to ko1 (Koichi Sasada)

I'm not sure why FCALL is dealt specially in op_send.
It doesn't seem needed if self is put instead of nil.

**#2 - 02/26/2012 04:23 AM - ko1 (Koichi Sasada)**

Hi,

(2012/02/11 6:56), Nobuyoshi Nakada wrote:

File 0001-Bug-5966-lambda-in-BasicObject.patch added
Status changed from Open to Assigned
Assignee set to Koichi Sasada

Thank you. Please commit it.

I'm not sure why FCALL is dealt specially in op_send.
It doesn't seem needed if self is put instead of nil.

It is a historical reason. No reason now.

--
// SASADA Koichi at atdot dot net

**#3 - 02/26/2012 11:26 AM - nobu (Nobuyoshi Nakada)**

- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r34819.
Kazuki, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- compile.c (iseq_compile_each): call on special object instead of self. since stabby lambda is a syntax, so it should not be affected by the context. [ruby-core:42349][Bug #5966]
- insns.def (send): no special deal for FCALL. self should be put on TOS instead.

**Files**

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001-Bug-5966-lambda-in-BasicObject.patch</td>
<td>2.42 KB</td>
<td>02/11/2012</td>
<td>nobu (Nobuyoshi Nakada)</td>
</tr>
</tbody>
</table>