Ruby master - Feature #6074
Allow alias arguments to have a comma
02/24/2012 08:31 PM - trans (Thomas Sawyer)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>matz (Yukihiro Matsumoto)</td>
</tr>
<tr>
<td>Target version</td>
<td>2.6</td>
</tr>
</tbody>
</table>

Description
This is one my biggest pet peeves with Ruby. I am always typing:

```
alias :foo, :bar
```

And getting a damn syntax error.

Btw, is there a reason why alias is a keyword and not a method?

History

#1 - 02/25/2012 07:41 AM - marcandre (Marc-Andre Lafortune)
- Priority changed from Normal to 3

If that this is your biggest pet peeve, you must really love Ruby!

Just use alias_method instead. The alias keyword is more general and allows other aliases like global variables; also you don’t need to give symbols to alias, you can write alias foo bar.

#2 - 02/25/2012 08:08 AM - trans (Thomas Sawyer)

Yes, I have almost exclusively used #alias_method in the past, but it’s always seems rather silly to have to use the longer term. And I bet it would be an easy adjustment to support a comma.

Btw, I really dislike using the non-symbol form of alias as it sticks out under syntax highlighters like a sore thumb.

#3 - 02/25/2012 09:53 AM - mame (Yusuke Endoh)

Hello,

2012/2/24 Thomas Sawyer transfire@gmail.com:

  Btw, is there a reason why alias is a keyword and not a method?

Just historical reason, I guess. A keyword-style alias was first introduced, and then alias_method was later introduced as a reflection API for the feature.

I understand the feeling. No keyword was needed for such a feature. If matz designed it now, he would implement it as a method, I believe.

  Btw, I really dislike using the non-symbol form of alias as it sticks out under syntax highlighters like a sore thumb.

In this regard, you should blame your syntax highlighter?

--
Yusuke Endoh mame@tsg.ne.jp

#4 - 02/25/2012 01:51 PM - ko1 (Koichi Sasada)
- Category set to core
- Target version set to 2.0.0

I think there are no reason why comma should be rejected.

03/16/2022
Hi,

(12/02/25 13:51), Koichi Sasada wrote:

I think there are no reason why comma should be rejected.

It's simple.

diff --git a/parse.y b/parse.y
index e47dac4..f55e754 100644
--- a/parse.y
+++ b/parse.y
@@ -979,40 +979,40 @@ stmt_or_begin : stmt
 %*/
 }

-stmt : keyword_alias fitem {lex_state = EXPR_FNAME;} fitem
+stmt : keyword_alias fitem opt_comma {lex_state = EXPR_FNAME;} fitem
 |
     /*%%%*/
-     $ = NEW_ALIAS($2, $4);
+     $ = NEW_ALIAS($2, $5);
     /*%
-     $ = dispatch2(alias, $2, $4);
+     $ = dispatch2(alias, $2, $5);
     %*/
 | keyword_alias tGVAR tGVAR
+ | keyword_alias tGVAR opt_comma tGVAR
 |
     /*%%%*/
-     $ = NEW_VALIAS($2, $3);
+     $ = NEW_VALIAS($2, $4);
     /*%
-     $ = dispatch2(var_alias, $2, $3);
+     $ = dispatch2(var_alias, $2, $4);
     %*/
 | keyword_alias tGVAR tBACK_REF
+ | keyword_alias tGVAR opt_comma tBACK_REF
 |
     /*%%%*/
 char buf[2];
 buf[0] = '$';
-     buf[1] = (char)$3->nd_nth;
+     buf[1] = (char)$4->nd_nth;
 $ = NEW_VALIAS($2, rb_intern2(buf, 2));
 /*%
-     $ = dispatch2(var_alias, $2, $3);
+     $ = dispatch2(var_alias, $2, $4);
     %*/
 | keyword_alias tGVAR tNTH_REF
+ | keyword_alias tGVAR opt_comma tNTH_REF
 |
     /*%%%*/
 yyerror("can't make alias for the number variables");
 $ = NEW_BEGIN(0);
 /*%
-     $ = dispatch2(var_alias, $2, $3);
+     $ = dispatch2(var_alias, $2, $4);
 $ = dispatch1(alias_error, $);%*/
}
@@ -5432,6 +5432,10 @@ opt_nl : /* none */
 | \n*/
 |
     +opt_comma : /* none */
+ | ',';
+ |
Nobu Nakada

#6 - 02/26/2012 04:13 AM - marcandre (Marc-Andre Lafortune)
I might have sounded more negative than I intended, so let me say that I also agree that we should allow a comma.

#7 - 02/28/2012 11:24 PM - shevegen (Robert A. Heiler)
I myself prefer "alias" because:

- It does not require a ','
- It is shorter to type
- It reads easier.

def bar
  puts 'Hi from bar.'
end

alias foo bar
alias_method :foo, :bar

The first way is more readable for my poor old eyes.

#8 - 03/30/2012 12:59 AM - mame (Yusuke Endoh)
- Status changed from Open to Assigned
- Assignee set to nobu (Nobuyoshi Nakada)

Sorry, I didn't know alias syntax accepts symbols as arguments!
Then I agree with this proposal.
Ruby always gives me a fresh surprise for me.

--
Yusuke Endoh mame@tsg.ne.jp

#9 - 11/20/2012 10:17 PM - mame (Yusuke Endoh)
- Target version changed from 2.0.0 to 2.6

#10 - 12/07/2012 11:12 PM - trans (Thomas Sawyer)
Is there any reason this can't make it into 2.0?

#11 - 12/08/2012 02:10 AM - alexeymuranov (Alexey Muranov)
Just another idea in this direction: allow

```ruby
def :foo, :bar do
  puts bar
end
```

in addition to

```ruby
def foo(bar)
  puts bar
end
```

(the "do" in the first case can be optional).

#12 - 12/08/2012 02:53 AM - drbrain (Eric Hodel)
On Dec 7, 2012, at 9:10, "alexeymuranov (Alexey Muranov)" redmine@ruby-lang.org wrote:

Issue #6074 has been updated by alexeymuranov (Alexey Muranov).

Just another idea in this direction:

If you have a separate idea open a separate feature request.
Do not hijack existing issues.
Hijacking issues makes them difficult for the committers to understand. Since this feature is delayed until next minor, continued off-topic discussion may lead to an issue being closed because its purpose is too difficult to determine.

#13 - 05/07/2014 06:15 AM - nobu (Nobuyoshi Nakada)
Salvaged. https://github.com/ruby/ruby/pull/609

#14 - 05/07/2014 06:18 AM - nobu (Nobuyoshi Nakada)
- Description updated

#15 - 05/07/2014 02:18 PM - nobu (Nobuyoshi Nakada)
- Assignee changed from nobu (Nobuyoshi Nakada) to matz (Yukihiro Matsumoto)

Request for syntax change.

#16 - 05/09/2014 02:05 AM - knu (Akinori MUSHA)
I believe the lack of comma is an important sign that alias is not really a method call in which directive arguments are not evaluated like with normal method calls.
If you are allowed to, and should you put a comma in an alias statement, it would look like a method call and confuse you in that you might think it could be passed any expressions.

Here is an example:

class Foo
  name = :to_str
  method = :to_s
  alias name, method # Likely a mistake, but won't err because `method` happens to exist.
end

I think we should instead deprecate the alias :sym1 :sym2 form so that it is clear that alias takes bare identifiers, not expressions.

#17 - 05/09/2014 02:34 AM - shugo (Shugo Maeda)
Akinori MUSHA wrote:

I believe the lack of comma is an important sign that alias is not really a method call in which directive arguments are not evaluated like with normal method calls.

Agreed, and this issue should be judged by Matz, shouldn't it?

#18 - 05/09/2014 03:14 AM - matz (Yukihiro Matsumoto)
- Status changed from Assigned to Rejected

As Akinori stated, alias is special to be special.
In your case, forget alias and always use alias_method.

Matz.

#19 - 05/14/2014 10:12 PM - trans (Thomas Sawyer)
Special cases always make me cringe. Why is alias a special case anyway? It seems very odd since alias_method is a method and does the same thing.