

Ruby trunk - Feature #6828

Constancy of Constants

08/04/2012 01:23 AM - trans (Thomas Sawyer)

Status:	Open
Priority:	Normal
Assignee:	
Target version:	Next Major
Description =begin Could have swore there was a recent issue addressing this, but couldn't find it, so... I hope by Ruby 2.0 module Foo::Bar::Baz will be equivalent to module Foo; module Bar; module Baz That constant lookup doesn't work the same for both is really really really annoying. =end	

History

#1 - 08/04/2012 04:39 AM - alexeymuranov (Alexey Muranov)

Did you mean [#6810](#) by any chance?

#2 - 08/15/2012 09:00 PM - MasterLambaster (Alex N)

```
=begin
I think it might be implemented but i'd rather not do that.
The reason why its not equivalent is because the first statement means:
((Define module Baz under Foo::Bar))
The second one means:
((Define module Foo, define module Bar under Foo, define module Baz under Foo::Bar))

So it's not an issue in a common sense. In first case you try to create module under namespace that does not exist, that leads to error, and I think that's as it should be. In the second case you you explicitly define all module structure.

=end
```

#3 - 08/15/2012 11:49 PM - trans (Thomas Sawyer)

```
=begin
It's not that. It's the constant lookup:
```

```
module Foo
X = 1
module Bar
p X #=> 1
end
end

module Foo::Bar
p X #=> NameError
end
```

Personally I would prefer undefined namespaces be automatically instantiated as modules. If that was not the intent another error will pick it up quickly enough anyway. But that's not the main issue --the constant lookup is the important thing.

```
=end
```

#4 - 08/15/2012 11:51 PM - trans (Thomas Sawyer)

[alexeymuranov \(Alexey Muranov\)](#) Yes, [#6810](#) that's the one. Probably why I did not find it --it listed as a Bug. I would agree, actually.

#5 - 08/16/2012 03:01 AM - drbrain (Eric Hodel)

- *Target version changed from 2.0.0 to Next Major*

=begin

I rely on this feature of constant lookup.

It allows me to define `File` and other top-level constants in my library's namespace, but still use the top-level `File` without littering `File::` everywhere in my library. It also helps me reduce confusion of contributors. They don't have to wonder if "File" is `File` or `MyLibrary::File`, it's always the top-level constant unless listed under my namespace. (In particular, RDoc has `RDoc::Encoding` and would break with this change.)

Changing the constant lookup will break ruby libraries that use this feature, so I have changed the target to 3.0

=end