

## CommonRuby - Feature #9941

### Issue a warning when `module` or `class` keyword causes re-initialization of a constant that will become the module/class name, instead of creating/reopening the module/class

06/13/2014 03:27 PM - alexeymuranov (Alexey Muranov)

<b>Status:</b>	Open	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Target version:</b>		
<b>Description</b>		
<p>I think that it would be appropriate to issue a warning similar to "warning: already initialized constant" when "module" or "class" keyword causes re-initialization of a constant. Here is an example of code that looks suspicious to me and IMO should issue such a warning:</p>		
<pre>module A   module B     C = 1   end end  module D   include A end  puts D.constants # =&gt; B puts D::B       # =&gt; A::B puts D::B::C    # =&gt; 1  module D::B end  puts D.constants # =&gt; B puts D::B       # =&gt; D::B puts D::B::C    # =&gt; uninitialized constant D::B::C (NameError)</pre>		
<p>P.S. I think this behavior is related to the impossibility to re-open anonymous classes/modules and to the somewhat strange IMO scoping rules for constants :).</p>		