

## Ruby master - Feature #7611

### Focal method for all loads/requires

12/23/2012 11:50 PM - trans (Thomas Sawyer)

<b>Status:</b>	Open
<b>Priority:</b>	Normal
<b>Assignee:</b>	matz (Yukihiro Matsumoto)
<b>Target version:</b>	
<b>Description</b>	
<pre>=begin Presently Ruby has seven methods for importing code. These are:</pre>	
<ul style="list-style-type: none"><li>• (<code>Kernel#load</code>)</li><li>• (<code>Kernel#require</code>)</li><li>• (<code>Kernel#relative_require</code>)</li><li>• (<code>Kernel.load</code>)</li><li>• (<code>Kernel.require</code>)</li><li>• (<code>Kernel.relative_require</code>)</li><li>• (<code>Kernel#autoload</code>)</li></ul>	
<p>Even though the Kernel module methods do the same thing as the instance methods, all of these act independently. If you need to tap into or override code loading in general it means doing so for each and every one.</p>	
<p>Would it not be much more elegant if they all called upon one focal method? Lacking another name for this example, lets call it (<code>Kernel#open_eval</code>). The method would take a path argument and options for (<code>feature</code>), (<code>relative</code>) and (<code>wrap</code>), where if (<code>feature</code>) is true then it is a "require", otherwise it is a "load".</p>	
<pre>def open_eval(path, feature: false, relative: false, wrap: nil) ... end</pre>	
<p>All the other seven methods listed above would route to this one method. So it would then be possible to monitor or override this behavior, e.g. like RubyGems does, via a single interface.</p>	
<p>Note, even if this feature request is not accepted, I would like to get some opinion on it, b/c I am currently working on a project where I have to do this (specifically I am in need of require/load callbacks). For the purpose I have created another reusable gem for it and I want to get an idea of what would be considered appropriate API for it.</p>	
<pre>=end</pre>	

### History

#### #1 - 12/24/2012 02:47 AM - zzak (Zachary Scott)

IRB adds even more requires, but I digress.

#### #2 - 02/22/2013 09:27 AM - ko1 (Koichi Sasada)

- Assignee set to matz (Yukihiro Matsumoto)

#### #3 - 12/25/2017 06:15 PM - naruse (Yui NARUSE)

- Target version deleted (2.6)