# Ruby master - Feature #16018
## Add a way to deprecate methods

07/24/2019 03:04 PM - vo.x (Vit Ondruch)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Open</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

Since [#11398](#11398), We have Module#deprecate_constant available. But I wonder, why we don't have a way to deprecate methods? There is already Gem::Deprecate [1], so may be this could be extracted and made more generic?

This might be useful on several places in Ruby code:

```bash
$ find . -name ".rb" -type f -exec grep -r "[^_]warn.*deprecate" {} \; | wc -l
```

66

**Related issues:**

Related to Ruby master - Feature #16345: Don't emit deprecation warnings by d...  
Closed

## History

### #1 - 07/24/2019 03:10 PM - jeremyevans0 (Jeremy Evans)
- Backport deleted (2.5: UNKNOWN, 2.6: UNKNOWN)
- ruby -v deleted (ruby 2.6.3p62 (2019-04-16 revision 67580) [i386-linux])
- Tracker changed from Bug to Feature

### #2 - 09/02/2019 06:24 AM - mame (Yusuke Endoh)

At the developers' meeting, no one was against this feature. It would be great if we can print a warning per a pair of callsite and callee (to suppress duplicated redundant warnings), but the implementation is not trivial (exploiting method call cache)?

### #3 - 12/06/2019 01:33 PM - vo.x (Vit Ondruch)
- Related to Feature #16345: Don't emit deprecation warnings by default. added