

## Ruby master - Feature #10720

### A proposal for something like: `attr_reader :foo?` - with the trailing '?' question mark

01/09/2015 02:23 AM - shevegen (Robert A. Heiler)

<b>Status:</b>	Rejected	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Target version:</b>		
<b>Description</b>		
<p>Hi guys,</p> <p>Sorry to burden you with another suggestion, no problem if it is not accepted, I wanted to just make it. :-)</p> <p>We have a way to define reader methods easily, such as in:</p> <pre>attr_reader :foo</pre> <p>This should be equivalent to:</p> <pre>def foo   @foo end</pre> <p>I love query methods ending with ?, so I often also use something like:</p> <pre>def foo?   @foo end</pre> <p>What if there is a programmatic way to define these?</p> <p>For instance, like:</p> <pre>attr_reader? :foo</pre> <p>This might lead to confusion perhaps, so a way such as:</p> <pre>attr_question :foo</pre> <p>or</p> <pre>attr_query :foo</pre> <p>or perhaps even better</p> <pre>attr_reader :foo?</pre> <p>The latter variant does not work:</p> <pre>foo.rb:20:in `attr_reader': invalid attribute name `foo?' (NameError)</pre> <p>Perhaps <code>:foo?</code> could query the instance variable <code>@foo</code> instead here? Currently <code>attr_reader :foo?</code> is invalid but <code>:foo?</code> is a valid Symbol in itself.</p> <p>At any rate, thanks for reading, may the year 2015 be great for all ruby devs and ruby users!</p>		
<b>Related issues:</b>		
Has duplicate Ruby master - Feature #11167: Allow an <code>attr_</code> variant for query-...		<b>Rejected</b>

## History

---

### #1 - 05/21/2015 11:02 PM - nobu (Nobuyoshi Nakada)

- Has duplicate Feature #11167: Allow an attr\_ variant for query-methods that end with a question mark '?' character, such as: `def foo?` returning `@foo` added

### #2 - 02/02/2016 04:55 AM - nobu (Nobuyoshi Nakada)

- Has duplicate Feature #12046: Allow attr\_reader :foo? to define instance method foo? for accessing @foo added

### #3 - 02/16/2016 06:54 AM - matz (Yukihiro Matsumoto)

- Status changed from Open to Rejected

See [#12046](#).

Matz.