**Description**

The "irb/completion" provides completion for classes, modules, symbols, and methods. This patch provides a new feature for it that shows documents when press TAB key one more after exact matches. This uses RDoc as a library.

**Example:**

```bash
$ ruby -Ilib bin/irb
```

```
irb(main):001:0> "".gsub # press TAB key twice
"".gsub "".gsub!
irb(main):001:0> "".gsub # press TAB key one more
# show RI document and press q to quit
irb(main):001:0> "".gsub
```

I'm the RDoc maintainer. A lot of Ruby users tell me "I'm using Google for reading documents of Ruby".

RDoc installs all documents to Ruby's directory by default. Many users never use it because it's just for RI("ri" command). I think that it is a reason of that many users don't attach importance to documentation.

Perl has "perldoc" feature and users easily access documents of modules by "perldoc" command. Python has "docstring" feature and users can access it on REPL. Those are each language's design of importance. Users use the language on the documentation design, so library developers write documents on the documentation design. Ruby doesn't have documentation design like Perl and Python. Ruby just has RDoc, IRB, and any other supports, but these are just fragmented features, these are not a documentation design.

Ruby needs many new features of documentation. This is one of them. This new feature gives new development environment.

**History**

### #1 - 05/25/2018 10:57 PM - shevegen (Robert A. Heiler)

I am fine with the suggestion.

If possible, a configuration option for IRB would be nice for people to decide on their own whether to use the above or not, but this is a small detail, not really that important anyway.

However had, to the comment:

RDoc installs all documents to Ruby's directory by default. Many users never use it because it's just for RI("ri" command). I think that it is a reason of that many users don't attach importance to documentation.

I remember many years ago, I was learning ruby through the old pickaxe, which was very nice; it provided cohesive documentation in a book format. I like books, they are a lot easier for me to read in a concentrated manner than computer screens. Anyway, back on topic, I was never very happy with ruby's online documentation. So not "just" rdoc but in general; that was a very long time ago though. Documentation has improved a lot but it's still not ... how shall I see it ... awesome.

Back then, a long time ago, I started to compile stuff on my own and store them in my local knowledgebase, which is unfortunately a mix of german and english. I usually prefer english since it is the language with more "leverage" (more people know it), but my brain simply processes german text faster/easier still.
I am, however had, also of the generation that can't deal with man-pages and instead simply "googles for wisdom"; and also uses Stack Overflow. When something seems useful, I integrate it into my local collection of useful stuff to know. Often I also find useful information on blogs, including about ruby.

When I compile ruby from source, which I do (I compile almost everything from source, using ruby for it of course; only the initial ruby-bootstrap may go via a shell script, to have ruby running), I always disable all documentation.

So no ri, rdoc, capi etc... I don't want any of these on my computer. Because of:

a) I won't need them.
b) I don't use them.
c) In particular the rdoc-generation step takes quite a long time, so skipping that makes compiling ruby faster; and sometimes bugs can happen where stuff fails, during that stage, which is not acceptable to me if I can avoid it.

Having said that, I rely on documentation of other ruby projects or projects in general; and I use markdown myself for gems I created and maintain. Markdown is really neat; I autogenerate README.md files based on a template which allows for some macros (commonly used things I have to do for some of my gems).

I also do not use "ri" on the commandline. I would not know why, because I myself simply do not use local look-up ways for documentation normally. I really "just google". And using the browser is about 100x more convenient for me as well.

The above is one use case. I am sure people have different means to use and work with documentation in general. But I am also sure that quite many people prefer browser-based documentation and tools at all times.

PS: Just before this is misunderstood, I agree with improving in-irb documentation in this regards. I love introspection, including things such as show-source in pry and so on.

#2 - 05/28/2019 12:06 AM - aycabta (aycabta .)
- Status changed from Open to Closed

Files

| show_doc_when_completion.patch | 8.9 KB | 05/25/2018 | aycabta (aycabta .) |