

Ruby - Bug #14728

OptionParser takes other options as argument values

05/01/2018 12:10 PM - xz0r (xz0r xz0r)

Status:	Rejected	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 2.0.0p648 (2015-12-16 revision 53162) [universal.x86_64-darwin16]	Backport: 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN
Description test.rb require 'optparse' options = {} OptionParser.new do opts opts.banner = "Usage: test.rb [options]" opts.on("--verbose", "Run verbosely") do v options[:verbose] = v end opts.on("--custom MANDATORY", "custom") do x options[:custom] = x end end.parse! p options If I run the above script as follows below is output: \$ ruby test.rb --custom --verbose {:custom=>"--verbose"} This is not the expected behaviour as --verbose should not be set as the option value.		

History

#1 - 05/01/2018 12:13 PM - xz0r (xz0r xz0r)

- Description updated

#2 - 05/01/2018 12:54 PM - nobu (Nobuyoshi Nakada)

- Description updated

- Status changed from Open to Rejected

If --custom with no argument is allowed, it is not "MANDATORY".

Try:

```
opts.on("-q", "--custom [CUSTOM]", "custom") do |x|
  options[:custom] = x
end
```

#3 - 05/02/2018 09:10 AM - xz0r (xz0r xz0r)

nobu (Nobuyoshi Nakada) wrote:

If --custom with no argument is allowed, it is not "MANDATORY".

Try:

```
opts.on("-q", "--custom [CUSTOM]", "custom") do |x|
  options[:custom] = x
```

```
end
```

--custom with no argument is not allowed, hence an error would be ideal in the above scenario.

#4 - 05/04/2018 10:29 AM - nobu (Nobuyoshi Nakada)

In common, an argument is determined by the preceding option, consider grep command's -e option.
If you want to disallow an argument starting with -, you'd need to specify a pattern

```
opts.on("-q", "--custom MANDATORY", "custom", /\A(?!-)/) do |x|
  options[:custom] = x
end
```

or allowed values

```
opts.on("-q", "--custom MANDATORY", "custom", %w[A B C]) do |x|
  options[:custom] = x
end
```

#5 - 05/11/2018 11:22 AM - xz0r (xz0r xz0r)

nobu (Nobuyoshi Nakada) wrote:

In common, an argument is determined by the preceding option, consider grep command's -e option.
If you want to disallow an argument starting with -, you'd need to specify a pattern

```
opts.on("-q", "--custom MANDATORY", "custom", /\A(?!-)/) do |x|
  options[:custom] = x
end
```

or allowed values

```
opts.on("-q", "--custom MANDATORY", "custom", %w[A B C]) do |x|
  options[:custom] = x
end
```

Thanks!