# Ruby - Feature #12350

## Introduce Array#find! that raises an error if element not found

05/05/2016 11:22 AM - ka8725 (Andrey Koleshko)

| | | |
|---|---|---|
| **Status:** | Open | |
| **Priority:** | Normal | |
| **Assignee:** | | |
| **Target version:** | | |

**Description**

It would be great to have some function (like Hash#fetch) for Array that raises an exception if element not found by passed block in find.

For example:

```
[].find! { |el| el == 1 }
=> ElementError: element not found
```

I just noticed that there are many cases when it's needed to raise an exception (because this is exceptional situation) if an element not found.

Currently I just define methods like  this one:

```
def find_price!(size)
  found_price = prices.find { |p| p.key == size }
  fail "Price for #{size} not found" unless found_price
  found_price
end
```

### History

**#1 - 05/05/2016 11:23 AM - ka8725 (Andrey Koleshko)**

*- Description updated*

**#2 - 05/05/2016 11:35 AM - zverok (Victor Shepelev)**

I'm pretty unhappy that community/style guidelines are banning the straightforward and readable solution:

```
prices.find { |p| p.key == size } or fail "Price for #{size} not found"
```

**#3 - 05/05/2016 01:30 PM - ka8725 (Andrey Koleshko)**

Victor Shepelev wrote:

> I'm pretty unhappy that community/style guidelines are banning the straightforward and readable solution:
>
> ```
> prices.find { |p| p.key == size } or fail "Price for #{size} not found"
> ```

I'm not aware about the banning. Could you elaborate what is bad in your example? Basically I don't see anything bad in your code except a little bit verbose like in mine solution.

**#4 - 05/05/2016 02:05 PM - zverok (Victor Shepelev)**

> Could you elaborate what is bad in your example? Basically I don't see anything bad in your code except a little bit verbose like in mine solution.

I don't know! The most popular and widely accepted Ruby Style Guide just bans them: https://github.com/bbatsov/ruby-style-guide#no-and-or-or

And when I've tried to rant about how and/or are useful, it all went sideways:
https://www.reddit.com/r/ruby/comments/478a17/rant_on_good_ruby_idiom_andor_operators/

**#5 - 05/05/2016 02:25 PM - nobu (Nobuyoshi Nakada)**

Victor Shepelev wrote:

> > Could you elaborate what is bad in your example? Basically I don't see anything bad in your code except a little bit verbose like in mine solution.
>
> I don't know! The most popular and widely accepted Ruby Style Guide just bans them: https://github.com/bbatsov/ruby-style-guide#no-and-or-or

I don't think it makes a sense, however, you can use || instead if you don't like or of course.

```
prices.find { |p| p.key == size } || raise "Price for #{size} not found"
```

And it feels better to me because of the error message.

#### #6 - 05/05/2016 02:46 PM - cremno (cremno phobia)

You can pass a callable object to Enumerable#find:

```
def find_price!(size)
  prices.find(->{ fail "Price for #{size} not found" }){ |p| p.key == size }
end
```

#### #7 - 05/20/2016 04:43 PM - dsferreira (Daniel Ferreira)

Documentation can be improved to use example of callable object don't you think?
I'm happy to create a PR with it.

#### #8 - 07/19/2016 06:20 AM - matz (Yukihiro Matsumoto)

I am not against adding a method that raises an exception, but find! is not a proper name for it.
Besides that, we can raise an exception for non-existing values by the current behavior.
So let us start improving documentation. Daniel, could you create an PR?

Matz.