## Ruby master - Feature #14606

**Change begin-else-end without rescue from warning to syntax error**

03/15/2018 07:59 AM - joker1007 (Tomohiro Hashidate)

<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

```ruby
begin
  p :foo
else
  p :bar
end
```

```ruby
# => :foo
# => :bar
```

```ruby
[1,2,3].each do
  p :foo
else
  p :bar
end
```

```ruby
# => :foo
# => :bar
```

begin-else-end without rescue is useless and dangerous. (especially, do-else-end is easy to mistake)

In actually, programmer never intend to write like these.

Ruby interpreter can guard this case by syntax error.

### History

#### #1 - 03/15/2018 01:51 PM - matz (Yukihiro Matsumoto)

Accepted.

Matz.

#### #2 - 03/16/2018 01:22 AM - mame (Yusuke Endoh)

I'm not against the removal.

BTW, surprisingly, the following code works "as intended."

```ruby
def fib(x)
  if x >= 2
    a = fib(x - 1)
    b = fib(x - 2)
    return a + b
  end else x
end
```

```ruby
p fib(10) #=> 55
```