Description

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

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

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

# => :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 removeal.

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

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

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