

## Ruby master - Feature #12995

### Conditional expression taking a receiver outside the condition

12/01/2016 07:00 AM - sawa (Tsuyoshi Sawada)

<b>Status:</b>	Rejected
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
<p>Since a conditional expression has a return value, we can continue a method chain after a conditional expression like this:</p>	
<pre>if hoge   foo1.foo2.foo3 else   bar1.bar2 end .baz1.baz2.baz3.baz4  case hoge when hoge1   foo1.foo2.foo3 when hoge2   bar1.bar2 end .baz1.baz2.baz3.baz4</pre>	
<p>It is convenient if we can have a condition after or in the middle of a chain. I would like to do this:</p>	
<pre>baz1.baz2.baz3.baz4. if hoge   foo1.foo2.foo3 else   bar1.bar2 end  baz1.baz2.baz3.baz4 if hoge   .foo1.foo2.foo3 else   .bar1.bar2 end  baz1.baz2.baz3.baz4. case hoge when hoge1   foo1.foo2.foo3 when hoge2   bar1.bar2 end  baz1.baz2.baz3.baz4 case hoge when hoge1   .foo1.foo2.foo3 when hoge2   .bar1.bar2 end</pre>	

#### History

#1 - 12/01/2016 12:15 PM - herwinw (Herwin Quarantainenet)

You can do that with #tap. A very stupid example:

```
array = [true]
res = array.to_a.tap do |obj|
  if obj[0]
    obj.replace(['true', :val])
  else
    obj.replace(['false', :val])
  end
end.to_a.to_h
p res
```

Results in:

```
{"true"=>:val}
```

**#2 - 02/22/2017 05:49 AM - matz (Yukihiko Matsumoto)**

*- Status changed from Open to Rejected*

Interesting proposal but this changes the meaning of if and case too much.

Matz.