
Subject: Re: Bug in SWITCH - ELSE: statement?

Posted by [Paul Van Delst\[1\]](#) on Thu, 11 Oct 2007 16:42:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

> Jean H wrote:

>>> You brought up something interesting (to me anyway.) If you place
>>> the

>>> ELSE earlier in the SWITCH statement, then everything below it is
>>> executed. If you do the same for a CASE statement then only then
>>> ELSE is executed. It never occurred to me until I saw your post
>>> that it was possible to place the ELSE anywhere but at the end.

>>

>> This is useless and can bring serious headaches...

>>

>> Both the Case and Switch run from the 1st condition to the last one...

>> so anything you write below the "else" will never (Case) or always

>> (Switch) be executed... then there is no need for a Case or Switch

>> anymore! ... just delete (Case) your code or write it outside of the

>> Switch block!

>

> Goodness.... I never even thought of that. Why on Earth should the order

> of the case selectors matter in a CASE construct? I just wrote:

>

> pro test_case, number

> case number of

> else: print, 'I do not like this number!'

> 1: print, '1 is a good number'

> 2: print, '2 is even better'

> 3: print, '3 is weird looking'

> endcase

> end

>

> IDL> .run test_case

> % Compiled module: TEST_CASE.

> IDL> test_case, 1

> I do not like this number!

> IDL> test_case, 3

> I do not like this number!

> IDL> test_case, 2

> I do not like this number!

> IDL> test_case, 1000

> I do not like this number!

>

> That is just ridiculous behaviour.

Ruby forces you to put it last:

```
#!/usr/bin/env ruby
number=ARGV[0].to_i
case number
  else
    puts("I don't like #{number}")
  when 1..3
    puts("#{number} is a good number")
end
```

```
Inx:scratch : ruby test_case.rb 5
test_case.rb:4: syntax error, unexpected kELSE, expecting kWHEN
test_case.rb:6: syntax error, unexpected kWHEN, expecting $end
  when 1..3
    ^
```

In the right order it works as expected:

```
#!/usr/bin/env ruby
number=ARGV[0].to_i
case number
  when 1..3
    puts("#{number} is a good number")
  else
    puts("I don't like #{number}")
end
```

```
Inx:scratch : ruby test_case.rb 1
1 is a good number
Inx:scratch : ruby test_case.rb 5
I don't like 5
```

Ideally I would prefer the order to not matter, but if it does then I think IDL should force the ELSE to be the last selector in CASE constructs (and probably SWITCH ones too)

cheers,

paulv
