Ruby master - Feature #10327

Bool/False/True module for '==='

10/05/2014 04:09 PM - arimay (yasuhiro arima)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		

Description

ruby-core:0237から始まるBoolean Classの話題でBoolean Moduleが提案されていました。
それは必要とされているのか？という疑問とともに.true?メソッドなどが議論されていました。

それを読んで、===メソッドを利用するBool, False, True Moduleを書いてみました。
これを使うとcase式でwhen TrueClass, FalseClassではなくwhen Boolと短く書けます。
また、case式で偽はwhen nil, falseではなくwhen Falseと書けます。
そして、真はelseではなくwhen Trueと書けます。
Bool, False, Trueという名前にしたのは、短く書きたいことと、リファレンスマニュアルでも使われているためです。

以下のrubyコードとほぼ同じ内容をCで書いたpatchを添えます。

```
module Bool
  def self.append_features(m)
    if ((m != FalseClass) and (m != TrueClass))
      raise TypeError, "`#{m.name}' isn't FalseClass nor TrueClass", caller(1)
    end
    super
  end
end

class FalseClass
  include Bool
end

class TrueClass
  include Bool
end

module False
  def self.===(other)
    other.nil? || other == false
  end
end
module True
  def self.===(other)
    !other.nil? && other != false
  end
end
```

以下が実行例です。

```
$ ./miniruby -e 'class String; include Bool; end;'
-e:1:in `append_features': 'String' isn't FalseClass nor TrueClass (TypeError)
        from -e:1:in `include'
        from -e:1:in `<class:String>'
        from -e:1:in `<main>'
$ ./miniruby -e '[nil, false, true, 0].each {|obj| p obj.class.ancestors}'
[NilClass, Object, Kernel, BasicObject]
[FalseClass, Bool, Object, Kernel, BasicObject]
[TrueClass, Bool, Object, Kernel, BasicObject]
[Fixnum, Integer, Numeric, Comparable, Object, Kernel, BasicObject]

$ ./miniruby -e '[nil, false, true, 0].each {|obj|
  case obj
```

03/07/2021 1/2

```
  when Bool; p [obj, Bool]
  else       p [obj, "-"]
  end
}'
[nil, "-"]
[false, Bool]
[true, Bool]
[0, "-"]

$ ./miniruby -e '[nil, false, true, 0].each {|obj|
  case obj
  when False; p [obj, false]
  when True;  p [obj, true]
  else        p [obj, "-"]
  end
}'
[nil, false]
[false, false]
[true, true]
[0, true]
```

History

#1 - 10/13/2014 10:05 AM - znz (Kazuhiro NISHIYAMA)

nilも含むならFalse/TrueよりもFalsy/Truthyの方が良いのではないでしょうか。

caseで使うだけなら:!.to_procと:itself.to_procを使うという方法もありそうです。
(itselfは2.2以降ですが。)

```
ruby -e '[nil, false, true, 0].each do |obj|
  case obj
  when :!.to_proc; p [obj, false]
  when :itself.to_proc; p [obj, true]
  else p [obj, "-"]
  end
end'
[nil, false]
[false, false]
[true, true]
[0, true]
```

#2 - 10/15/2014 02:42 PM - arimay (yasuhiro arima)

- File bool-falsy-truthy.patch added

もっともなのでFalsy/Truthyに変えてみました。
patchを作り直したので添付します。
前回のpatchは壊れていました。確認が不十分でごめんなさい。

```
$ ./miniruby -e '[nil, false, true, 0].each {|obj|
  case obj
  when Falsy;  p [obj, false]
  when Truthy; p [obj, true]
  else         p [obj, "-"]
  end
}'
[nil, false]
[false, false]
[true, true]
[0, true]
```

また、to_procを使う案ですが、短く書きたいのが主題なのでそれでは嬉しくないのです。

Files

bool-false-true.patch	3.44 KB	10/05/2014	arimay (yasuhiro arima)
bool-falsy-truthy.patch	3.85 KB	10/15/2014	arimay (yasuhiro arima)

03/07/2021 2/2