



Learning Object-Oriented
Programming and Design with TDD

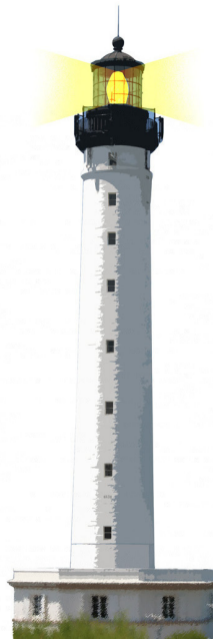
Variables

Stéphane Ducasse

<http://stephane.ducasse.free.fr>



<http://www.pharo.org>



In a Nutshell

- **Local** variables start with **lowercase** (temps, instance variables, arguments,...)
- **Shared** variables start with **uppercase** (class, class variables)

Temporary variables are local to the method

Example: c

```
CounterTest >> testIncrement  
  |c|  
  c := Counter new.  
  ...
```

Remember: class names start with uppercase

Local Variables Start with Lowercase

Instance variables are local to the object

Example: x, y, count

```
Object subclass: #Point  
  instanceVarNames: 'x y'
```

```
Object subclass: #Counter  
  instanceVarNames: 'count'
```

Local Variables Start with Lowercase

Method arguments: aPoint

```
crossProduct: aPoint
```

"Answer a number that is the cross product of the receiver and the argument, aPoint."

```
^ (x * aPoint y) - (y * aPoint x)
```

Block arguments: :x

```
[ :x | x + 2 ]
```

Special Variables

Special variables cannot be changed

- true, false, nil
- self, super, thisContext

Special Variables

- true, false are the Booleans
 - true is the unique instance of the class True
 - false is the unique instance of the class False
- nil is the unique instance of the class UndefinedObject

Special Variables

- `self` refers to the receiver of the message (`this` in Java)
- `super` refers to the receiver but the method lookup starts in the superclass of the class defining the method (see dedicated Lectures)
- `thisContext` refers to the current execution stack (advanced)

Shared or Global Variables start with Uppercase

Object is a class globally accessible

```
Object subclass: #Point
```

Transcript is an object that is globally accessible (a kind of stdout)

```
Transcript cr.  
Transcript show: 'hello world'.  
Transcript cr.
```


ClassVariables are Shared Variables

- To share information between all the instances of a class and subclasses
- Use a classVariable

```
Object subclass: #CombinedChar  
  instanceVariableNames: 'codes combined'  
  classVariableNames: 'Compositions Decompositions Diacriticals'  
  package: 'Kernel-BasicObjects'
```

- Here Compositions is shared between all the CombinedChar instances and instances of subclasses

Resources

- Pharo mooc - Videos W3S03: <http://mooc.pharo.org>
- Pharo by Example: <http://books.pharo.org>

Summary

- Lowercase are used for local/private/temporary variables
- Uppercase are used for shared or global variables

A course by Stéphane Ducasse
<http://stephane.ducasse.free.fr>

Reusing some parts of the Pharo Mocc by

Damien Cassou, Stéphane Ducasse, Luc Fabresse
<http://mocc.pharo.org>



Except where otherwise noted, this work is licensed under CC BY-NC-ND 3.0 France
<https://creativecommons.org/licenses/by-nc-nd/3.0/fr/>