Variables

Damien Cassou, Stéphane Ducasse and Luc Fabresse
In a Nutshell

- **Local** variables start with `lowercase` (temps, instance variables, arguments,...)
- **Shared** variables start with `uppercase` (class, class variables)
Local Variables Start With Lowercase

Temporary variables are local to the method
Example: c

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

Remember: class names start with uppercase
Instance variables are local to the object
Example: x, y, count

Object subclass: #Point
instanceVarNames: 'x y'

Object subclass: #Counter
instanceVarNames: 'count'
**Method arguments:** aPoint

```plaintext
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

```plaintext
[ :x | x + 2 ]
```
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)
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
Summary

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

and

in collaboration with

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/