Prof. Dr. Oscar Nierstrasz
Pascal Gadient, Pooja Rani

Solution
Assignment 02 — 26.09.2018 – v1.0e
Smalltalk: A Reflective Language

Please submit this exercise by mail to sma@list.inf.unibe.ch before 03 October 2018, 10:15am.

Exercise 1 - Nature of Smalltalk and Pharo (3 Points)

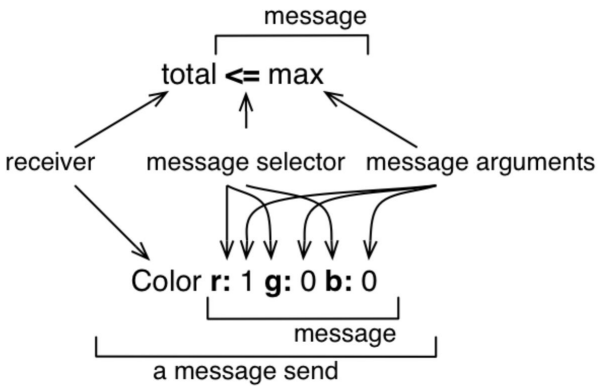
a) Name three major benefits Pharo provides in comparison to Java. **Answer:**

- *Pharo provides intercession that can change the state of components during run time.*
- *Everything can be changed including system objects and language features.*
- *Pharo is a live environment instead of IDEs for Java that just execute code on demand (manual recompilation requests).*

b) How can you export the current image of your Pharo working environment? **Answer:**

The image state can be saved persistently with a left mouse click on the workbench, followed by "Save" or "Save As".

c) What is a message in Pharo? **Answer:**



All communication or interaction among objects is done by sending messages. A message is composed of the message selector and the optional message arguments. A message is sent to a receiver. The combination of a message and its receiver is called a message send.

d) What is a block in Pharo? **Answer:**

A block can be thought of as a lambda-expression defining an anonymous function, or as a function object. Square brackets `[ ]` define a block, also known as a block closure or a lexical closure, which is a first-class object representing a function.
Blocks may take one or more arguments (`[:i :j :k | ... ]`) and can have local variables.

Exercise 2 - Pharo object inspection (4 Points)

a) Which message is being sent to a superclass to find all its subclasses? **Answer:**

subclasses

b) What is the difference between a `String` and a `Symbol` object in Pharo? Why is this differentiation important?
Hint: The execution of the code below will reveal some of the differences.

```
('HeySmalltalker') == 'HeySmalltalker'.
('HeySmalltalker') asSymbol == #HeySmalltalker.
('Hey','Smalltalker') == 'HeySmalltalker'.
('Hey','Smalltalker') asSymbol == #HeySmalltalker.
```

Answer:

- *Symbols are immutable and unique. Strings are mutable and not unique. Consequently, multiple String objects with the value "Hello" can coexist, but only one Symbol object #Hello can be initialized during run time. Since Symbols are immutable, they should never be used when their value has to change over time. The benefit of using Symbols is the performance gain due to less expensive value comparisons.*
- *('HeySmalltalker') == 'HeySmalltalker'.*
  Comparison of two Strings, expected to return true (note that the outcome depends on the VM implementation).

  ('HeySmalltalker') asSymbol == #HeySmalltalker.
  Comparison of two Symbols, expected to return true.

  ('Hey','Smalltalker') == 'HeySmalltalker'.
  Comparison of two (concatenated) Strings, expected to return false (note that the outcome depends on the VM implementation).

  ('Hey','Smalltalker') asSymbol == #HeySmalltalker.
  Comparison of two (concatenated) Symbols, expected to return true.

c) How can you create an abstract method in Pharo? **Answer:**

Implementation of the following code in the class body:

```
sampleAbstractMethod
    ^self subclassResponsibility.
```

d) Write the equivalent of the following piece of code in Smalltalk as a block, and execute it with the values <38, 44>, <65, 48>, and <48, 48>. Please use the *Transcript tool* available in Pharo to

retrieve the output.

```
int scoreOfPlayerA, scoreOfPlayerB;
if(scoreOfPlayerA > scoreOfPlayerB)
    print "Player A Won"
else if(scoreOfPlayerA < scoreOfPlayerB)
    print "Player B Won";
else
    print "Match is declared as draw";
```

Answer:

Please download the solution here.

Exercise 3 - Even more Pharo coding (3 Points)

Execute Smalltalk code to answer the questions below about the demo application you can download here.
Hint: You need to extend the CallGraph.st class with some additional accessors.

Don't forget to submit your code *and* your results.

a) Find the top 10 most frequently invoked methods.

b) Find the top 10 methods invoked on the largest number of different classes (dynamic types).

c) Find all static methods.

Answer:

Please download the solution here.