



Learning Object-Oriented Programming and Design with TDD

About this lectures: OO Programming and OODesign

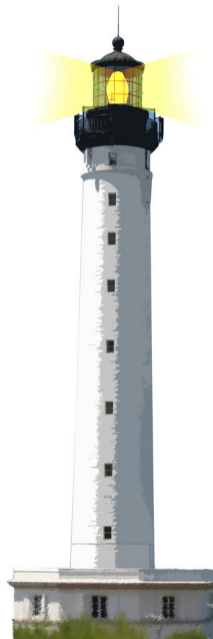
S. Ducasse and S. Costiou

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



<http://www.pharo.org>

Core



A word of Presentation

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

- Core developer of Pharo - Head of the consortium
- Language design (traits in PHP, Scala, Perl, Groovy...), maintenance, tests,...
- Expert in OO Design, OO implementation and more

Steven

- 6 years of experience in industry
- Agile development (SCRUM, TDD)
- Expert in debuggers/OOP/OOD



Objectives

- Become comfortable with object-oriented programming: Learn to think with objects
- Learn some Java (but not deeply because this is too complex)
- Tests
- Essence of OOP:
 - what really means to send a message
 - what is this/super
 - some limited design patterns



About this lectures

- Not a lecture on how to program
- A lecture on how to DESIGN programs
- It is not enough just to write code that works
- Design starts when the program compiles



Give some solid foundation for ANY OOP language

The world is complex, there are many languages

- Give you a key survival guide
- Concepts available in any languages



What is design?

- You learned so far how to code: writing functions, algorithms, structure ideas
- Making extensible, maintainable, readable code
 - Tests
 - Design nice and beautiful objects
 - Some simple design patterns



Our teaching challenges

- Some of you already 'know' (got exposed to) Java
- Some never really programmed in OOP
- Java is complex and ugly (it polluted the essence of objects)

Solution: Exposing yourself to beauty

We do not grow 'Chefs' by feeding them with McDonald

- Showing Java
 - explaining some selected points
- Comparing with Pharo
- Provide core and more advanced lectures (free access/consumption)



Solution: A la carte

- Everybody should know a bit of Java at the end
- A possibility
 - follow the Pharo Mooc: <http://mooc.pharo.org> and do the exercises there
 - You will get +4 bonus on exam for your investment
- if you want to learn more you can do both :)



Pharo / Mooc

- I have already learned so much ! I have spent the last 20 years or so in software development and, following this Mooc, I realized I hadn't really grasped the essence of object oriented design. Anonymous
- "I just completed the @pharoproject Mooc the best investment I have ever made of my time. MAQBOOL"
- - Watch <http://rmod-pharo-mooc.lille.inria.fr/MOOC/PharoMOOC-Videos/C019SD-Teaser-v4.mp4>
 - Syntax fit on a postcard
 - Fully immersive
 - Will change totally the way to think about objects
 - learn XtremeTDD



Outline

- Tests
- Java in a nutshell
- Rethinking message passing
- Rethinking `this/ super / Inheritance`
- Hooks and template
- Sending messages are plans for reuse
- Avoid is null
- Inheritance vs. Composition



At the end of the lecture

You should be able to

- Write tests and understand why but you need to practice
- Write small programs in Java / Pharo
- Understand late binding for real
- Get first level about what good design is about



About laptops...

<https://cs.brown.edu/courses/cs019/2018/laptop-policy.html>



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://mooc.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/>