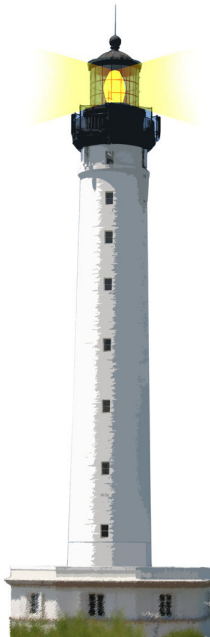
Learning Object-Oriented Programming and Design with TDD

Type aspects of Java

S. Ducasse

http://stephane.ducasse.free.fr



Pharo

http://www.pharo.org

Objectives

- Understanding dynamic and static
- Casts

A type of a variable

Let us a simple program model:

- a variable is a box with a label, its type.
- a variable contains references to objects.

A variable type indicates the kind of object the variable can refer to

```
A a
```

In the variable `a` we can put reference to object of the class A (and more see after)

Static vs. Dynamic Types

```
A a = new B();
```

- The static type of variable a is A i.e., the statically declared class to which it belongs.
  - The static type never changes.
- The dynamic type of a is B i.e., the class of the object currently bound to a.
  - The dynamic type may change throughout the program.

```
a = new A();
```

Now the dynamic type is also A!

Static vs. Dynamic Types

This works too with method signature

```
public class A { }
public class B extends A { }
```

```
public class Main {
  public static void main(String[] args) {
    dynclassOutput (new B());
    dynclassOutput (new A());
  }
  public static void dynclassOutput (A a) {
    System.out.println(a.getClass().getName());
  }
}
```

What is the static / dynamic type of a there?

The Key question

What should be relationship between A and B in the following to be valid?

```
A a = new B();
```

Example

```
class Rectangle {}
class Box extend Rectangle {}
class ColoredBox extend Box {}
```

```
Box b = new Box();
Box b = new ColoredBox()
>>>Valid!
```

```
Box b = new Rectangle()
>>> invalid
```

Why because the program may use b.volume() and in rectangle there is such volume() method.

Overloading is a bad idea

You can have multiple methods with the same name and types argument

```
visit (ProgramNode n) {}

visit (Assignment n) {}

visit (SequenceNode n) {}
```

- Avoid it as much as possible... it makes code less extensible
- Overloading makes your code difficult to understand in presence of subtyping
- Programming in OOP is using subtyping
- Check other lectures

Kind of Summary

- Static types are known by the compiler.
- Dynamic types are the variable values known at execution.

A course by Stéphane Ducasse

http://stephane.ducasse.free.fr

Reusing some parts of the Pharo Mooc by

Damien Cassou, Stéphane Ducasse, Luc Fabresse

http://mooc.pharo.org