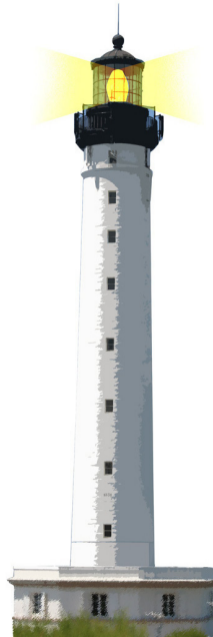


Legends and cheatsheets

S. Ducasse



<http://www.pharo.org>



Class definitions

Pharo

```
Rectangle subclass: #Box  
instanceVariableNames: 'height'  
classVariableNames: ''  
package: 'MyPackage'
```

```
Rectangle << #Box  
slot: { #height};  
package: 'MyPackage'
```

Java

```
class Box {  
    protected int height;  
    ...  
}
```



Constructors

```
Box >> initialize  
  super initialize.  
  height := 100
```

```
public Box() {  
  super();  
  height = 100;  
}
```



```
SetTestCase >> setUp  
  super setUp.  
  empty := Set new.
```

```
@Before  
public void setUp(){  
  empty = new Set();  
}
```

Test methods

```
SetTestCase >> testAdd  
empty.add: 5. "Stimulus"  
empty.add: 5.  
self assert: empty size equals: 1 "Check"
```

```
@Test  
public void testAdd(){  
    empty.add(5); //Stimulus  
    empty.add(5);  
    assertEquals(empty.size(), 1); "Check"  
}
```

Syntactic elements

	Pharo	Java
comment	"a comment"	\\ or /* */
character	\$c \$# \$@	
string	'lulu' 'l"idiot'	"lulu"
symbol (unique string)	#mac #+	
literal array	#(12 23 36)	[int]
boolean	true, false	
undefined	nil	null
point	10@120	

Syntactic elements II

	Pharo	Java
temp declaration	temp	int temp ;
return	^ true	return true
assignment	x := 12	x = 12
separator	x := 12 .	x = 12 ;

Essential constructs

- Messages

```
1 class  
1 + 2  
self send: aMail to: stef
```

```
1.getClass()  
1 + 2  
this.sendTo(aMail,stef)
```

- Lexical closure definition and execution

```
[ :x | x + 2 ] value: 5  
>>> 7
```



A course by

S. Ducasse, G. Polito, and Pablo Tesone



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/>