

OOP goals		OOP building blocks (cont)		OOP Paradigms (c	
Simple modeling	An accurate representation of the real world by grouping objects with their properties and actions	Objects	An instance of a class. (the class is the type definition and the object is the variable) => all instances of a class share the same fields but have distinct data inside.	Heritage	A des relatic shorte ties/ e own c
Robustness	Easy maintenance and bug detection. Strong typing which results in more robust code (predictable behavior of your code)		In addition to its state (properties) and behavior (methods) an object has an identity which distincts it from other objects (alias/ memory address ...)		Compos- ition, Agregation
Scalability	Adding functionality comes down to establishing new connections with other objects and methods.				
Reusability	Available features allowing code reusability. (inheritance)				
OOP building blocks		OOP Paradigms		JAVA general info	
Class	A blueprint of an object: its properties, behavior and how it interacts with the exterior world.	Abstra- ction	The selection of only useful information about an object for a particular application.	JAVA platform = JV	
Attributes & methods	The attributes are the properties of an object while the methods represents its dynamic behavior inside of your program. These attributes and methods define how the object should be accessed, its internal behavior and how it interacts other objects		Encaps ulation	Is the ability to control the access to the object's properties and methods, render them either visible (public) or hidden (visible only to internal functions).	JVM : execution en Allows code to be n executes inside of a abstracts the spece configuration, and c JVM has its own na like a real computer The JVM interprets memeory for the pr Garbage collector
		Polymo rphisme		The definition of different executions of a methode for different inputs and for different objects	JAVA API : libraries onalities.
				JAVA main functio	
				<pre>public class test { public static void main (String args[]) { // code } }</pre>	
				Array declaration	
				<p>In java you must all by</p> <pre>type [] name=new name[] = new ty</pre> <p>Then if the type is n tiate each field of th array_name[i]=n attributes)</p> <p>Also, we can declar type array_name</p>	



Strings

Declaration:

```
String s="sample text";
String s=new String("sample text")
```

Some string methods:

```
s.equals(s2); returns 1 is s=s2 and 0 otherwise
s3=s1.concat(s2); <=> s3=s1+s2; to concatenate
strings
```

I/O

INPUT in the header of the class file :

```
import java.util.Scanner;
```

in the method:

```
Scanner sc = new Scanner(System.in);
int i = sc.nextInt();
double d = sc.nextDouble();
long l = sc.nextLong();
byte b = sc.nextByte(); // etc
```

OUTPUT for single variables:

```
System.out.println(single variable);
```

for multiple variables:

```
System.out.println(var1+" "+va-
r2+...);
```

the idea is to convert variables to strings and concatenate them

Garbage Collection

```
//The garbage collector deallocates memory of
unreferenced objects
//example
cl obj1=new cl();
System.gc();
//the GC does nothing here since obj 1 is
still referenced
cl=null;
```

Garbage Collection (cont)

```
System.gc();
//the GC deallocates the memory previously allocated for obj
referenced
//This makes the life of a programmer (for example when free
objects : a single command instead of a loop
```

Properties of static class members

Static attributes are initialized as follows: ints, floats -> **0**, booleans -> **false**, references -> **null**. Static methods have access only to static methods and attributes of a class (obviously not **this** reference since it doesn't make any sense).

```
Class_name.static_attribute_name
Class_name.static_method_name()
```

Class members visibility

Modificateur	classe	package	Sous-classe	Autres packages
public	oui	oui	oui	oui
protected	oui	oui	oui	non
package-private	oui	oui	non	non
private	oui	non	non	non



By **mahdi007**
cheatography.com/mahdi007/

Published 26th October, 2021.
 Last updated 26th October, 2021.
 Page 2 of 2.

Sponsored by **Crossw**
 Learn to solve cryptic c
<http://crosswordcheats>