

# Programmation Orienté Objet

## - JAVA -

### Chapitre 1

#### Programmation Orienté Objet – Notions de Base

[bekkalimohammed@gmail.com](mailto:bekkalimohammed@gmail.com)

# POO – Notions de base

## Plan

I. La programmation classique

II. Inconvénients de la programmation classique

III. La POO

IV. Concepts de base de la POO

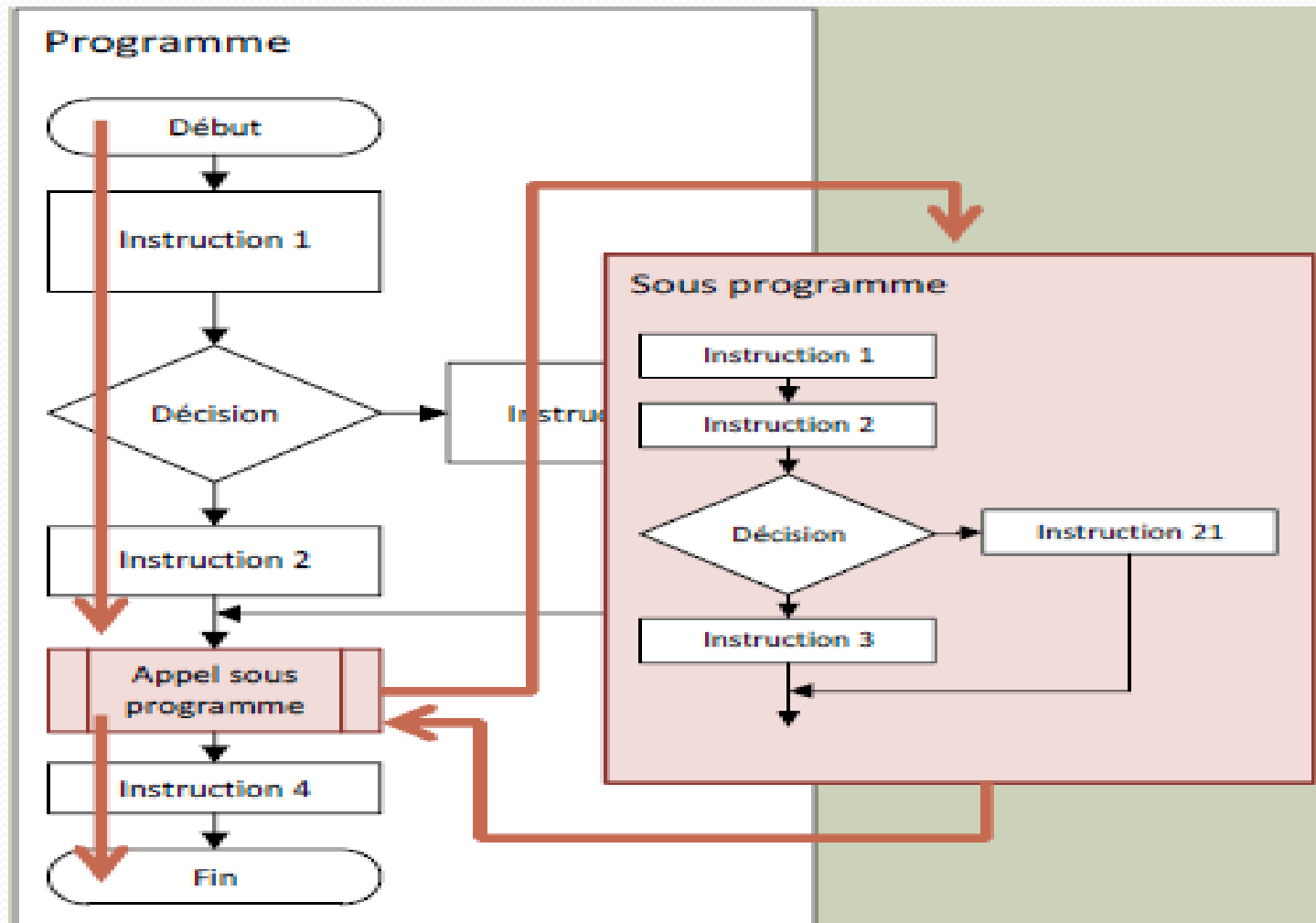
- Encapsulation des données
- Abstraction
- Classe et Objet
- Héritage
- Polymorphisme

# I. La programmation classique

La programmation procédurale telle que étudiée au travers des langages C, Pascal... définit un programme comme étant un ensemble de données sur lesquelles agissent des procédures et des fonctions.

Cette approche a permis de décomposer les fonctionnalités d'un programme en procédures qui s'exécutent séquentiellement .

# I. La programmation classique



# I. La programmation classique

Dans ce cas,

Programmer → 1- définir un certain nombre de variables (structures, tableaux...)  
2- écrire des procédures pour les manipuler sans associer explicitement les unes aux autres.

Exécuter → appeler ces procédures dans un ordre décrit par le séquençage des instructions et en leur fournissant les données nécessaires à l'accomplissement de leurs tâches.

## II. Inconvénients de la programmation classique

Devant la complexité des applications, cette technique de programmation présentait quelques inconvénients :

- Cette méthode oblige le développeur à penser de manière algorithmique.
- Le développeur doit faire un effort supplémentaire pour structurer le programme.
- Séparation entre les procédures et les données.

## II. Inconvénients de la programmation classique

Suite,

- Le code est peu lisible et difficile à modifier, à maintenir
- L'ajout de fonctionnalités difficile
- Réutilisation du code incertaine
- Le travail d'équipe est délicat

## II. Inconvénients de la programmation classique

- Finalement, il est peut-être plus simple de s'inspirer du monde réel
  - ✓ Le monde réel est composé d'objets, d'êtres vivants, de matière
  - ✓ Pourquoi ne pas programmer de manière plus réaliste ?
- Les objets ont des propriétés
  - ✓ Un chat à 4 pattes, un serpent aucune
- Les objets ont une utilité, une ou plusieurs fonctions
  - ✓ Une voiture permet de se déplacer
  - ✓ Un couteau peut couper
- Alors, plutôt que de focaliser sur les procédures, intéressons-nous d'avantage aux données
- De cette analyse est née la programmation orientée objet



# III. La Programmation Orienté Objet

Définition:

La POO est une méthode d'implémentation dans laquelle les programmes sont organisés sous formes de collections coopératives d'objets, dont chacun représente une instance d'une classe quelconque et dont toutes les classes sont membres d'une hiérarchie de classes unis à travers des relations d'héritage.

# III. La Programmation Orienté Objet

Qu'est-ce qu'un objet ?

- Un élément qui modélise toute entité, concrète ou abstraite, manipulée par le logiciel
- Un élément qui réagit à certains messages qu'on lui envoie de l'extérieur
- Un élément qui ne réagit pas toujours de la même manière

L'objet est l'élément central de la POO

# III. La Programmation Orienté Objet

## Objet et classe

Un objet est une entité logicielle :

- Ayant une identité
- Capable de sauvegarder un état c'est-à-dire un ensemble d'information dans des variables internes.
- Répondant à des messages précis en déclenchant des activations internes appropriés qui changent l'état de l'objet. Ces opération sont appelées méthodes. Ce sont des fonctions liées à des objets et qui précisent le comportement de ces objets.

# III. La Programmation Orienté Objet

## **Attributs**

Les attributs d'un objet sont l'ensemble des informations se présentant sous forme de variable et permettant de représenter l'état de l'objet.

## **Message**

Un message est une demande d'activation d'une méthode envoyé à un objet.

## **Méthode**

Une méthode est une fonction ou procédure liée à un objet qui est déclenchée à la réception d'un message particulier

## **Signature**

La signature d'une méthode représente la précision de son nom, du type de ses arguments et du type de donnée retournée.

## IV. Concepts de base de la POO

La POO se base sur les notions clés suivantes :

- Encapsulation
- Abstraction
- Classe et objets
- Héritage
- Polymorphisme

## IV. Concepts de base de la POO

### **Encapsulation des données:**

C'est le fait de réunir à l'intérieur d'une même entité (objet) le code (méthodes) + données (champs). Il est donc possible de masquer les informations d'un objet aux autres.

On dit qu'il y a encapsulation de données et du comportement des objets.

## IV. Concepts de base de la POO

### Encapsulation des données:

On précise trois modes d'accès aux attributs d'un objet.

- Le mode **public** avec lequel les attributs seront accessibles directement par l'objet lui-même ou par d'autres objets. Il s'agit du niveau le plus bas de protection.
- Le mode **private** avec lequel les attributs de l'objet seront inaccessibles à partir d'autres objets : seules les méthodes de l'objet pourront y accéder. Il s'agit du niveau le plus fort de protection.
- Le mode **protected** avec lequel les attributs seront accessibles directement par l'objet lui-même ou par ses objets descendants (lien d'héritage).

## IV. Concepts de base de la POO

### **Abstraction:**

C'est le fait de se concentrer sur les caractéristiques importantes d'un objet selon le point de vue de l'observateur.

L'abstraction est un principe qui consiste à ignorer certains aspects d'un sujet qui ne sont pas importants pour le problème dans le but de se concentrer sur ceux qui le sont.



## IV. Concepts de base de la POO

### Classe et objet:

Avant de créer des objets, il faut définir un modèle et des objets pourront être créés à partir de ce modèle.

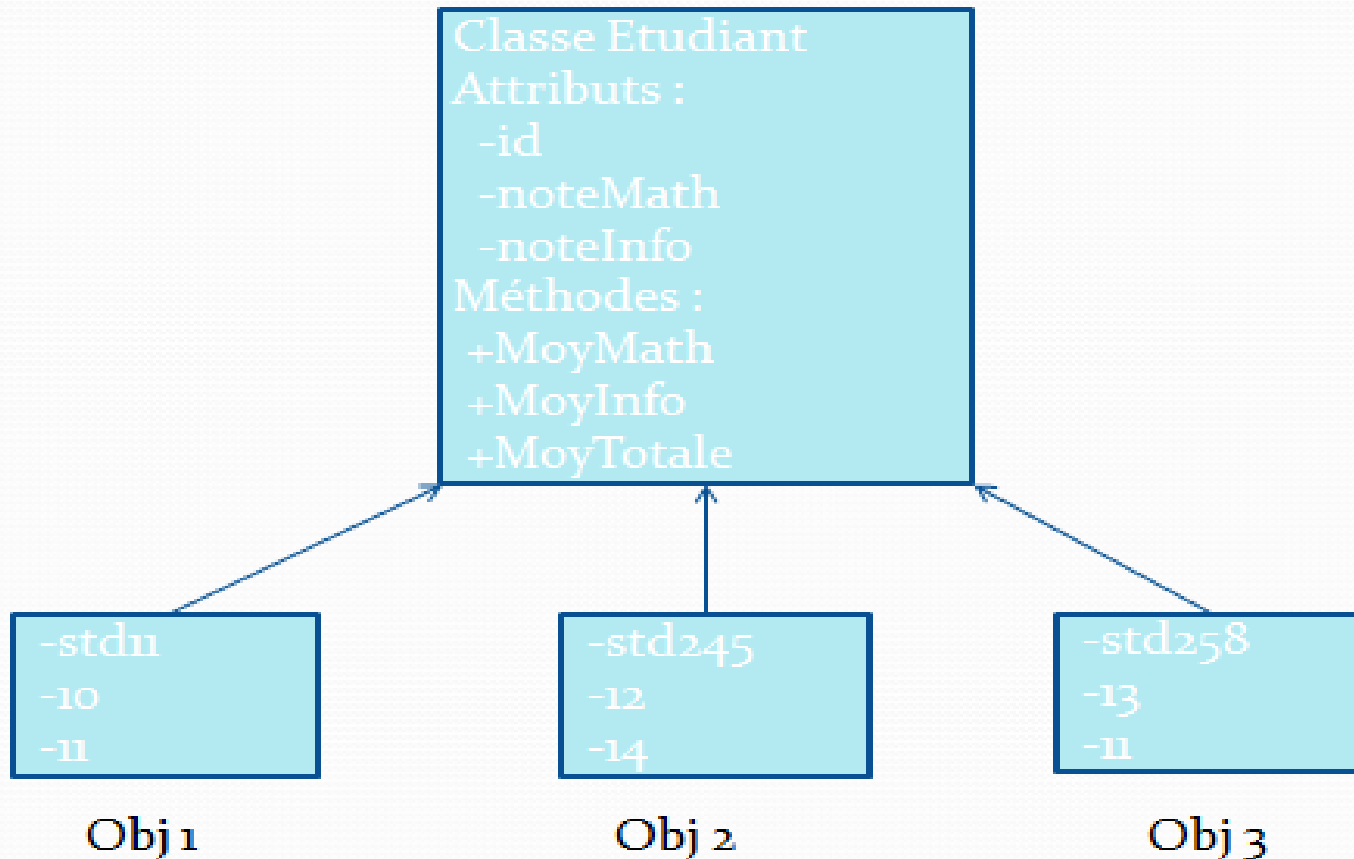
Ce modèle s'appelle une **classe** et les objets fabriqués à partir du modèle sont des **instances**.

Une Classe est composé:

- D'attribut (ou champs, ou variables d'instances) et ils décrivent la structure de ses instances (les objets).
- De méthodes (ou opérations de la classe) décrivant les opérations applicables aux instances de la classe.

## IV. Concepts de base de la POO

### Classe et objet: Exemple

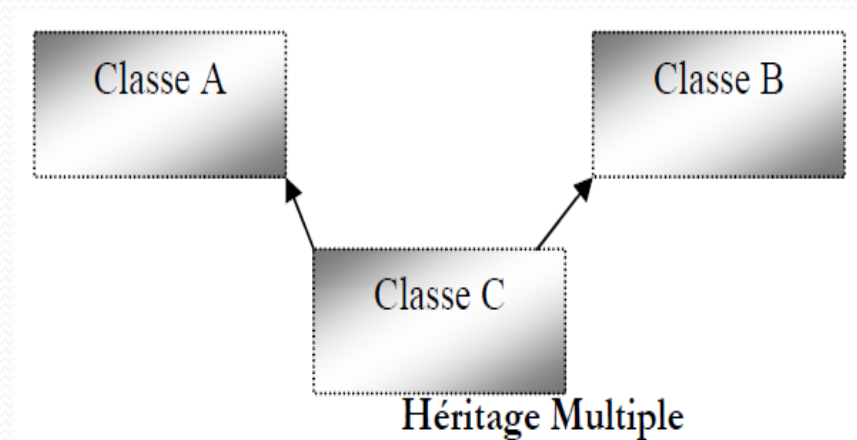
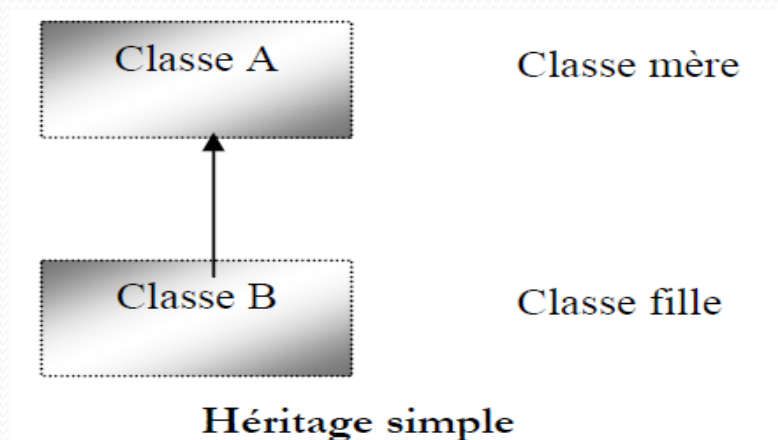


## IV. Concepts de base de la POO

### Héritage:

La notion d'héritage est une relation entre différentes classes permettant de définir une nouvelle classe en se basant sur les classes existantes. On parle d'héritage simple lorsqu'une classe fille ne possède qu'une classe mère.

On parle d'héritage multiple lorsqu'une classe fille possède plusieurs classes mères.



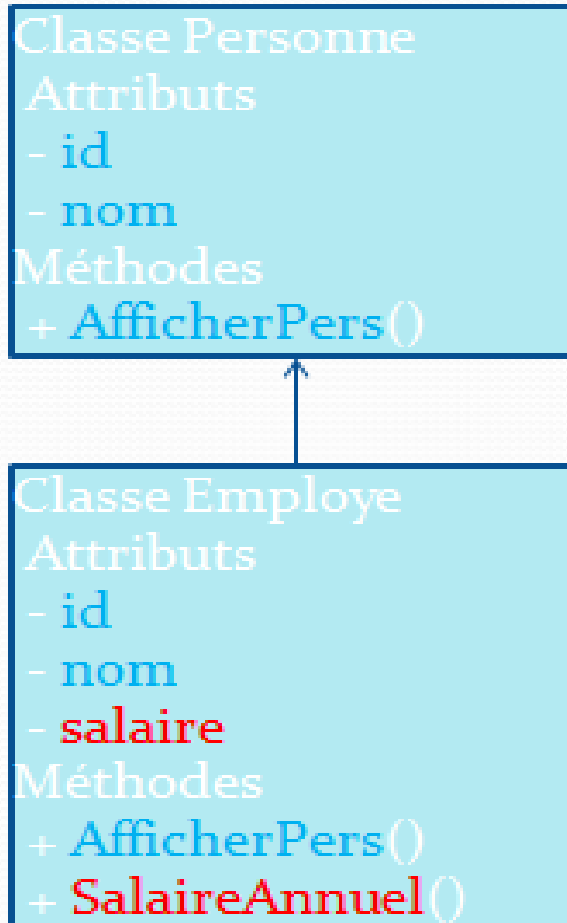
## IV. Concepts de base de la POO

### **Héritage:**

- Par construction toutes les sous-classes d'une même classe possèdent toutes les attributs et les méthodes de leur classe mère.
- Les attributs et les méthodes peuvent être modifiés au niveau de la sous-classe.
- Il peut y avoir des attributs et/ou des méthodes supplémentaires dans une sous-classe

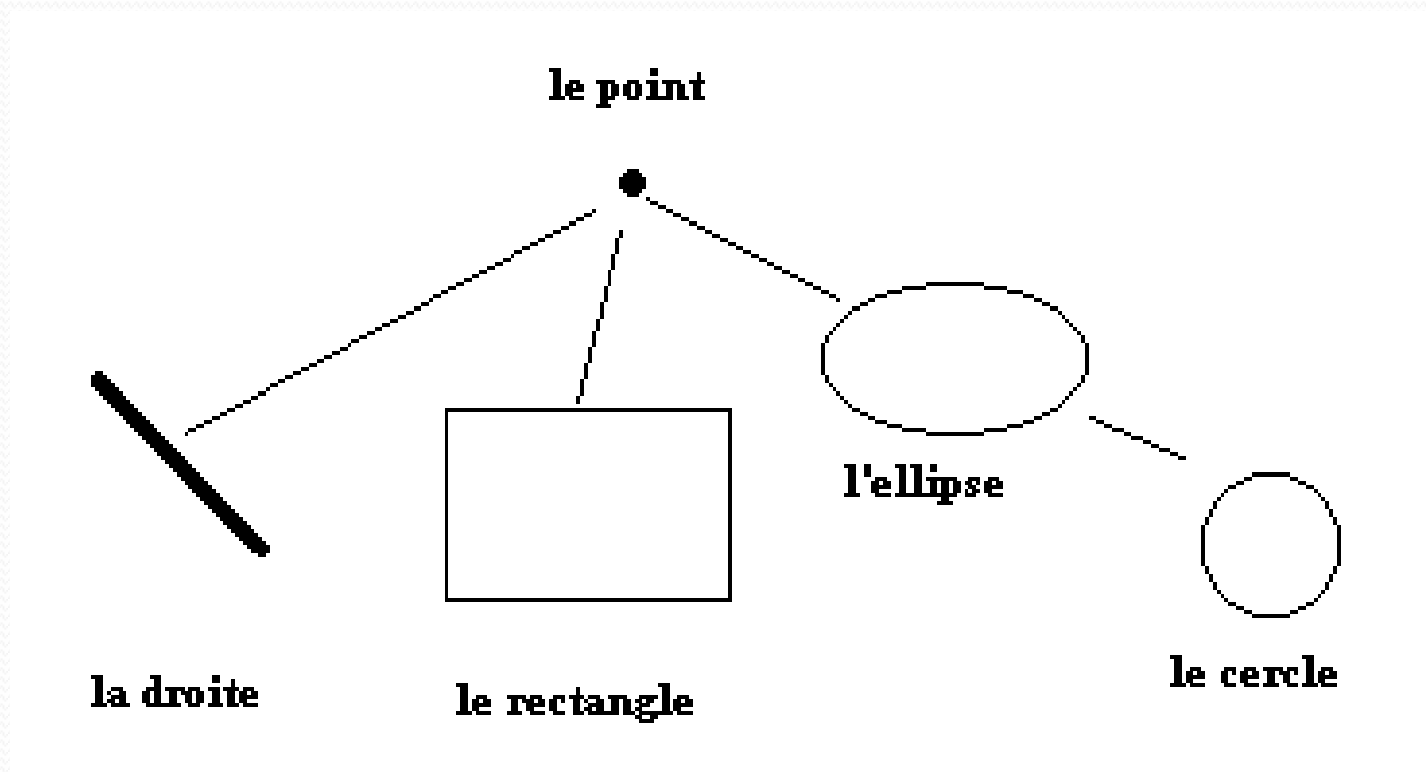
## IV. Concepts de base de la POO

### Héritage: Exemple



## IV. Concepts de base de la POO

### Héritage: Exemple



## IV. Concepts de base de la POO

### **Polymorphisme:**

Le terme polymorphisme issu du grec signifie la possibilité de prendre plusieurs formes.

Une entité est polymorphe si à l'exécution elle peut se référer à des instances de classe différentes.

