

TP numéro 3

Objectifs

- Utilisation des objets à l'intérieur des classes.
- Héritage.
- Redéfinition.
- Polymorphisme.
- Abstraction.

Exercice 1

Le but de l'exercice est de concevoir une application qui sera utilisé au sein de la faculté. Pour cela, il faut développer un ensemble de classes comme suit :

Personne : Une personne contient un nom, un prénom, une adresse et un numéro de téléphone. La classe comprend également une méthode d'affichage qui affiche toutes les informations d'une personne sur une seule ligne.

Employe : Employe hérite de Personne. Un Employe comprend également :

- un numéro de sécurité sociale et un nom de département ;
- une méthode d'affichage qui affiche toutes les informations d'un employé.

Professeur : Professeur hérite de Employe. Cette classe comprend également :

- un champ booléen qui indique si le professeur est responsable d'un module ;
- une méthode d'affichage qui affiche toutes les informations d'un professeur.

Depratement : Cette classe comprend :

- un nom ;
- un chef de département (c'est un professeur).

Etudiant : Etudiant hérite de Personne. En plus des attributs de Personne, un étudiant contient :

- un CNE et il est inscrit dans une filière ;
- une méthode d'affichage qui affiche toutes les informations d'un étudiant.

- Les différents attributs des différentes classes doivent être privés.
- Si nécessaire, ajoutez des setters et de getters ;
- Dans chaque classe, ajoutez deux constructeurs : un par défaut et l'autre initialise les différents attributs.
- Mettez chaque classe dans un fichier séparé.

1. Écrivez une application nommée GestionFaculte qui déclare :

- un employé ;
- deux professeurs ;
- deux étudiants ;

- un département.
- 2. Demander à l'utilisateur de spécifier le type de données de la personne qui sera entré : e (employé), p (professeur), t (étudiant), ou permettre à l'utilisateur de quitter (q).
- 3. Lorsque l'utilisateur quitte, affichez les informations concernant les différentes personnes. Si l'utilisateur n'a pas saisi de données pour un ou plusieurs types de personnes, affichez un message approprié.

Exercice 2

1. Créez une classe **Personne** qui contient :
 - les attributs **privés** :
 - **nom**, **prenom** de type **String**;
 - **age** de type **int**;
 - **admin** de type boolean (true pour administrateur et false pour client).
 - un constructeur pour initialiser les différents attributs;
 - une méthode abstraite nommé **setAdmin**;
 - une méthode **afficher()** qui affiche le nom, le prénom et l'âge.
2. Créez une classe **CompteBancaire** qui doit posséder :
 - les attributs privés **numCompte** de type **int** et **solde** de type **double**;
 - un constructeur pour initialiser les différents attributs;
 - la méthode **deposer(double s)** : pour ajouter une somme au solde;
 - la méthode **retirer(double s)** : pour retirer une somme du solde;
 - la méthode **info()** qui retourne les informations concernant le compte, comme suit :
« compte numéro : 121, Solde : 15000 Dhs ».
3. Créez une classe **Client** qui hérite de la classe **Personne** :
 - on suppose que chaque client dispose d'un seul compte;
 - redéfinissez la méthode **afficher** pour qu'elle affiche aussi les informations concernant le compte.

Indication : ajoutez à la classe **Client** un attribut de type **CompteBancaire** qui sera initialisé dans un constructeur.
4. Écrivez une classe **Banque** qui contient une méthode **main()**. Dans la méthode **main()** :
 - a - déclarez 3 clients qui auront les caractéristiques suivantes :

	Nom	prénom	Age	Solde initial
Client 1	Oujdi	Ali	30	1000Dh
Client 2	Berkani	Lina	27	2000Dh
Client 3	Figuigui	Karim	35	3000Dh

Pour le numéro de compte, le premier client aura le numéro **1000** et pour chaque nouveau client il faut ajouter **1** au numéro du client précédent (ajoutez dans la classe **client** un attribut statique qui sera utilisé pour incrémenter le numéro de compte).

- b - Le client 1 dépose 1000 Dh.
- c - La cliente 2 retire 1500 Dh.
- d - Affichez les informations concernant les différents clients.