

TP 03-POO

Objectif :

L'objectif de ce TP est de permettre à l'étudiant :

- a. De manipuler les concepts suivants :
 - Classe, objet, instanciation,
 - Constructeurs
 - Surcharge de méthodes
 - Encapsulation
- b. De distinguer les:
 - Variable d'instance et méthode d'instance
 - Variable de classe et méthode de classe
 - Constante et son initialisation
- c. Affichage avec l'instruction `System.out.println()`
- d. De faire la saisie à partir du clavier avec la classe `Scanner`
- e. Affichage de toutes les caractéristiques d'un objet avec la méthode `afficher()` et des `System.out.println()`
- f. Affichage de toutes les caractéristiques d'un objet en redéfinissant la méthode `toString()`

Étape de création des classes `CompteBancaire` et `AppliCompteBancaire` sous Eclipse

1. Au niveau du projet **TP03POO**
2. Créer un nouveau package dans **File/New/Package**, nommer ce package **exercice1**
3. Créer une nouvelle classe dans **File/New/Class**, nommer cette classe **CompteBancaire**
4. Créer une nouvelle classe dans **File/New/Class**, nommer cette classe **AppliCompteBancaire**
5. Cocher la case: **public static void main(String[] args)**

Implémenter le code de la classe `CompteBancaire` en suivant les indications suivantes :

Nous définissons la classe `CompteBancaire` avec les attributs et méthodes suivante :

1. Attributs :

- `numCompte`: un nombre entier représentant le numéro du compte;
- `nomClient` : une chaîne de caractère mémorisant le nom du client ;
- `prenomClient` : une chaîne de caractère mémorisant le prénom du client ;
- `solde`: un nombre en double précision conservant le solde du compte ;
- `nomBanque` : une chaîne caractère représentant le nom de la banque, exemple : **"AL-BARAKA"**
- `codeAgence` : une chaîne de caractère représentant le code d'une agence particulière, exemple : **"AGENCE-CODE-107"**
- `typeCompte` : une chaîne de caractère mémorisant le type du compte qui peut être soit :
compteCourant, compteEpargne ;

Remarque : le nom de la banque et le code de l'agence sont des variables communes à tous les comptes, par contre le type de compte est propre à chaque compte mais n'est pas modifiable. Nous souhaitons aussi connaître le nombre de compte, pour cela apporter les modifications requises.

2. Constructeurs

On dotera cette classe des deux constructeurs avec paramètres avec des valeurs initiales explicites pour les attributs et un autre constructeur sans paramètres, initialisant les attributs aux valeurs initiales.

3. Accesseurs

- Ajouter les accesseurs (getters et setters) nécessaires.

4. Méthode d'instance :

Nous nous contentons pour l'instant que de la méthode d'affichage.

- void afficher(): affiche à l'écran toutes les caractéristiques de l'état du CompteBancaire;

5. Méthode de classes :

Ajouter une ou des méthodes de classes si nécessaire.

Ecrire le code de la méthode main de la classe

Partie 01 :

- Déclarer deux références du compte sur un CompteBancaire : *compte1* et *compte2*;
- Créer/construire/initialiser le compte1 avec le constructeur à paramètres ;
 - numCompte = 1 ;
 - nomClient = "Bouakkaz";
 - prenomClient = "Samra";
 - solde= 10000.00
 - typeCompte = "CompteEpargne"
- Créer/construire/initialiser le compte2 avec le constructeur à paramètres:
 - numCompte = 2 ;
 - nomClient = "Aberbour";
 - prenomClient = "Rima" ;
 - solde= 10000.00 ;
 - typeCompte = "CompteCourant"
- afficher compte1 et compte2
- afficher le nombre de comptes créés
- Proposer une solution pour que la variable *numCompte* s'incrémente automatiquement (à l'instanciation). Apportez les modifications requises et faire un test en instanciant et en affichant deux autres instances compte3 et compte4.

Partie 02 : Utilisation de la classe Scanner

- Lire à partir du clavier les données nécessaires pour construire le compte5 avec un constructeur à paramètres.
- Afficher compte5
- Déclarer et construire un compte6 avec le constructeur sans paramètres
- Lire à partir du clavier les données nécessaires pour modifier les valeurs des variables de compte6
- Proposer une solution dans le cas d'erreur.
- Afficher compte6

Partie 03 : tableau de *CompteBancaire*:

Nous exploiteront le TP03 pour faire un rappel sur les tableaux en Java et la notion de packages

1. Au niveau du projet **TP03POO**
2. Créer un nouveau package dans **File/New/Package**, nommer ce package **exercice02**
3. Créer une nouvelle classe dans **File/New/Class**, nommer cette classe **AppliCompteTab**
4. Cocher la case: **public static void main(String[] args)**

Implémenter le code de la classe *AppliCompteTab* en suivant les indications suivantes :

- Importer les packages nécessaires
- Déclarer et créer un tableau de *CompteBancaire* en lisant la taille à partir du clavier,
- Lire le tableau de *CompteBancaire* (remplissage à partir du clavier),
- Affichage du tableau de *CompteBancaire* (avec un for classique et un for étendu).

Remarque :

- Importer la classe *CompteBancaire* « **import exercice1.CompteBancaire** »
- Importer la classe *Scanner* pour la lecture à partir du clavier
- Modifier l’affichage de classe *Compte* en redéfinissant la méthode **toString()**

Partie 04 : Redéfinition de la méthode **toString()**

- **String toString()**: redéfinir la méthode **toString ()** pour afficher à l’écran toutes les caractéristiques de l’état du *CompteBancaire* (au niveau de la classe *CompteBancaire*).
- Tester l’affichage avec la méthode **toString()** (au niveau de la classe *AppliCompteBancaire*).

Bon Travail