



مكتب التكوين المهني وإنعاش الشغل

Office de la Formation Professionnelle et de la Promotion du Travail  
Direction Régionale Tensift Atlantique

Etablissement : ISTA NTIC SYBA

Formateur : FAKHOURI

EFM : PROGRAMMATION Orientée Objets  
2010/2011

Filière: TECHNIQUES DE DEVELOPPEMENT INFORMATIQUE

Groupe(s) : D & B

Niveau : 1<sup>ère</sup> année

Durée : 1h30

Barème: /40

On souhaite modéliser la gestion des comptes bancaires d'un établissement financier grâce à la programmation orientée objet.

Un client est repéré par un numéro de client, une date d'adhésion et les comptes qu'il possède. Et ils sont de deux catégories Client physique (nom, date de naissance) et Client moral (raison sociale, forme juridique).

Un compte bancaire est caractérisé par son numéro, sa date d'ouverture, son solde. Outre la création et la fermeture d'un compte, on peut effectuer des versements et des retraits :

Créditer un compte consiste à ajouter un montant positif au solde du compte.

Débiter un compte consiste à retirer un montant positif au solde du compte. Le solde résultant ne doit en aucun cas être inférieur à zéro.

Travail demandé

- 1- Créer une classe Compte avec les attributs (numéro : entier, date d'ouverture : date, solde : Double) **(1pt)**
- 2- Ajouter un constructeur par défaut (le numéro est un nombre générer aléatoirement composé de 5 chiffres, la date d'ouverture est initialisée à la date d'aujourd'hui et le solde à 0) **(2pts)**
- 3- Ajouter un constructeur d'initialisation (le numéro est un nombre générer aléatoirement composé de 5 chiffres, la date d'ouverture est initialisée à la date d'aujourd'hui et le solde est initialisé à une valeur passée en paramètre) **(1pt)**
- 4- Ajouter les propriétés nécessaires **(2pts)**
- 5- Ajouter une méthode crediter(montant : double) **(1pt)**
- 6- Ajouter une méthode debiter(montant : double), lever une exception SoldeInsuffisantException si le montant à retirer est supérieur au solde du compte **(3pts)**
- 7- Créer une classe abstraite Client (numéro de client, une date d'adhésion et les comptes qu'il possède) **(2pts)**
- 8- Ajouter un constructeur par défaut et un autre d'initialisation. **(2pts)**
- 9- Ajouter les propriétés nécessaires. **(2pts)**
- 10- Ajouter la méthode equals() pour tester l'égalité de deux clients selon le numéro. **(2pts)**
- 11- Créer une classe ClientPhysique qui hérite de la classe Client et définit deux nouveaux attributs (nom, date de naissance) **(2pts)**

- 12- Ajouter les constructeurs par défaut et d'initialisation, ainsi que les propriétés nécessaires.  
Lever une exception AgeInvalideException si l'âge du client est inférieur à 18ans. **(4pts)**
- 13- Créer une classe ClientMoral qui hérite de la classe Client et définit deux nouveaux attributs  
(raison sociale, forme juridique : SARL ou SA) **(2pts)**
- 14- Ajouter les constructeurs par défaut et d'initialisation, ainsi que les propriétés nécessaires.  
Lever une exception JuridiqueInvalideException si la forme juridique est incorrecte. **(4pts)**
- 15- Ajouter une méthode ajouterCompte permettant d'ajouter un compte à la liste des  
comptes du client **(1pt)**
- 16- Ajouter une méthode getCompteParNumero(numero) as Compte pour renvoyer le  
compte dont le numéro est passé en paramètre. **(2pts)**
- 17- Ajouter une méthode getComptesParDate(d1 as date, d1 as date ) as List( Of  
Compte) pour renvoyer les comptes créer entre d1 et d2 **(3pts)**

Question 1	1 point
Question 2	1 point
Question 3	2 points
Question 4	2 points
Question 5	1 point
Question 6	3 points
Question 7	2 points
Question 8	2 points
Question 9	2 points
Question 10	2 points
Question 11	2 points
Question 12	4 points
Question 13	2 points
Question 14	4 points
Question 15	1 point
Question 16	2 points
Question 17	3 points
Bonne rédaction et comportement en examen	4 points

Formateur : FAKHOURI	Directeur Pédagogique	Directeur du complexe/Directeur de l'EFP	Visa de La DRTA

*Bon courage*