

Dossier SI4.4 Exo2 : les Classes

Exercice 1 - La Classe Livre :	1
Question 1 :	1
Question 2 :	1
Question 3 :	3
Question 4 :	3
Question 5 :	3
Exercice 2 : Classe Compte :	4
Question 1 :	4
Question 2 :	4
Question 3 :	4
Question 4 :	5
Question 5 :	5
Question 6 :	5
Question 7:	5
Question 8 :	6
Question 9 :	6
Question 10 :	8
Code :	9

Exercice 1 - La Classe Livre :

Question 1 :

```
public class Livre {  
    //Attributs  
    private String titre;  
    private String auteur;  
    private int prix;
```

Question 2 :

```
    // Constructeur  
    public Livre(String titre, String auteur, int prix) {  
        this.titre = titre;  
        this.auteur = auteur;  
        this.prix = prix;  
    }
```

```
    //Constructeur qui demande les valeurs à l'utilisateur  
    public Livre() {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.print(s: "Entrez le titre du livre : ");  
        this.titre = scanner.nextLine();  
  
        System.out.print(s: "Entrez l'auteur du livre : ");  
        this.auteur = scanner.nextLine();  
  
        System.out.print(s: "Entrez le prix du livre : ");  
        this.prix = scanner.nextInt();  
  
        System.out.println(x: "Livre créé avec succès !");  
    }
```

Question 3 :

```
// Getters
public String getTitre() {
    return titre;
}

public String getAuteur() {
    return auteur;
}

public int getPrix() {
    return prix;
}

// Setters
public void setTitre(String titre) {
    this.titre = titre;
}

public void setAuteur(String auteur) {
    this.auteur = auteur;
}
```

Question 4 :

```
//Méthode Afficher
public void Afficher() {
    System.out.println("Titre : " + titre);
    System.out.println("Auteur : " + auteur);
    System.out.println("Prix : " + prix + " €");
}
```

Question 5 :

```
public static void main(String[] args) {
    Livre livre = new Livre();

    livre.Afficher();
}
```

Exercice 2 : Classe Compte :

Question 1 :

```
import java.util.Scanner;
// Classe Client
class Client {
    private String CIN;
    private String Nom;
    private String Prenom;
    private String Tel;
```

Question 2 :

```
// Getters
public String getCIN() {
    return CIN; }

public String getNom() {
    return Nom; }

public String getPrenom() {
    return Prenom; }

public String getTel() {
    return Tel; }
```

Je pars du principe qu'il n'est pas nécessaire de définir de setters car d'après moi il n'y a pas à changer le CIN, le Nom, le Prénom ... du titulaire du compte car ce sont des informations personnelles qui ne changent pas (peut être le numéro de téléphone mériterait un setters)

Question 3 :

```
// Constructeur (initialisation totale)
public Client(String CIN, String nom, String prenom, String tel) {
    this.CIN = CIN;
    this.Nom = nom;
    this.Prenom = prenom;
    this.Tel = tel;
}
```

Question 4 :

```
// Constructeur partiel (initialisation uniquement de CIN, nom et prénom)
public Client(String CIN, String nom, String prénom) {
    this(CIN, nom, prénom, tel: "");
}
```

Question 5 :

```
// Méthode Afficher()
public void Afficher() {
    System.out.println("CIN: " + CIN);
    System.out.println("NOM: " + Nom);
    System.out.println("Prénom: " + Prénom);
    System.out.println("Tél : " + Tel);
}
```

Question 6 :

```
class Compte {

    private static int increment = 1;

    private int numeroCompte;
    private float solde;
    private Client proprietaire;
```

Question 7:

```
// Getters
public int getNumeroCompte() {
    return numeroCompte; }

public float getSolde() {
    return solde; }

public Client getProprietaire() {
    return proprietaire; }
```

Question 8 :

```
// Constructeur
public Compte(Client proprietaire) {
    this.numeroCompte = increment++;
    this.solde = 0f;
    this.proprietaire = proprietaire;
}
```

Question 9 :

Méthodes de Crédit :

```
public void Crediter(float montant) {
    solde += montant;
    System.out.println(x: "Opération bien effectuée");
}

public void Crediter(float montant, Compte source) {
    if (source.solde >= montant) {
        source.solde -= montant;
        this.solde += montant;
        System.out.println(x: "Opération bien effectuée");
    } else {
        System.out.println(x: "Solde insuffisant !");
    }
}
```

Méthodes de Débit :

```
public void Debiter(float montant) {
    if (solde >= montant) {
        solde -= montant;
        System.out.println(x: "Opération bien effectuée");
    } else {
        System.out.println(x: "Solde insuffisant !");
    }
}

public void Debiter(float montant, Compte destination) {
    if (solde >= montant) {
        solde -= montant;
        destination.solde += montant;
        System.out.println(x: "Opération bien effectuée");
    } else {
        System.out.println(x: "Solde insuffisant !");
    }
}
```

Méthode d’Affichage :

```
// Méthode Afficher (résumé d'un compte)
public void Afficher() {
    System.out.println(x: "*****");
    System.out.println("Numéro de Compte: " + numeroCompte);
    System.out.println("Solde de compte: " + solde);
    System.out.println(x: "Propriétaire du compte : ");
    proprietaire.Afficher();
    System.out.println(x: "*****");
}
```

Question 10 :

```
// --- Programme principale ---
public class CompteProgramme {

    Run | Debug
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println(x: "Compte 1:");
        System.out.print(s: "Donner le CIN: ");
        String cin = sc.nextLine();

        System.out.print(s: "Donner Le Nom: ");
        String nom = sc.nextLine();

        System.out.print(s: "Donner Le Prénom: ");
        String prenom = sc.nextLine();

        System.out.print(s: "Donner le numéro de téléphone: ");
        String tel = sc.nextLine();

        Client C1 = new Client(cin, nom, prenom, tel);
        Compte cp1 = new Compte(C1);

        cp1.Afficher();

        System.out.print(s: "Donner le montant à déposer: ");
        float m1 = sc.nextFloat();
        cp1.Crediter(m1);

        cp1.Afficher();

        System.out.print(s: "Donner le montant à retirer: ");
        float m2 = sc.nextFloat();
        cp1.Debiter(m2);

        cp1.Afficher();

        sc.close();
    }
}
```


Code :

<https://drive.google.com/file/d/10C-Lx8npCzd6rm3ubxY-m6v0knkdYKKU/view?usp=sharing>