

## Solution Examen Programmation Orientée Objet

### Exercice 1 (10p)

Soit le code suivant :

```
1. class Livre
2. {
3.     private String titre;
4.     private String auteur;
5.     private float prix;
6.     private int quantite;
7.
8.     public Livre()
9.     {
10.    }
11.
12.    public Livre(String t, String a, float p, int qt)
13.    {
14.        titre = t;
15.        auteur = a;
16.        prix = p;
17.        quantite = qt;
18.    }
19.
20.    public void afficher()
21.    {
22.        System.out.println("Titre : "+titre+", Auteur : "+auteur+", Prix : "+prix+", Qté : "+quantite);
23.    }
24.
25.    public void setTitre (String titre)
26.    {
27.        this.titre = titre;
28.    }
29.
30.    public String getTitre()
31.    {
32.        return titre;
33.    }
34. }-----
35.
36. public class ExamenEx1
37. {
38.     public static void main(String[] args)
39.     {
40.         Livre x = new Livre("Object Programing", "Said",120 , 10);
41.         x.afficher();
42.         Livre y = new Livre();
43.         y.setTitre("Necluar Physic");
44.         System.out.println("Titre : "+y.getTitre());
45.     }
46. }-----
```

#### Questions :

Quels sont les attributs de la classe Livre : (1)... titre ..... (2)... auteur ..... (3)..... prix ..... (4)... quantite .....

Donner 4 méthodes de la classe Livre : (1)..... Livre()..... (2)... afficher()..... (3)..... setTitre ..... (4)..... getTitre().....

Définir le constructeur plusieurs fois (dans les lignes 8 et 12), ceci est appelé :  Redéfinition  Surdéfinition

Qu'affiche la ligne 41 : ..... Titre : Object Programing, Auteur : Said, Prix : 120.0, Qté : 10.....

Ecrire la méthode setTitre() qui permet de définir le titre d'un livre. Répondre dans les lignes 25 et 27.

Ecrire la méthode getTitre() qui retourne le titre d'un livre. Répondre dans les lignes 30 et 32.

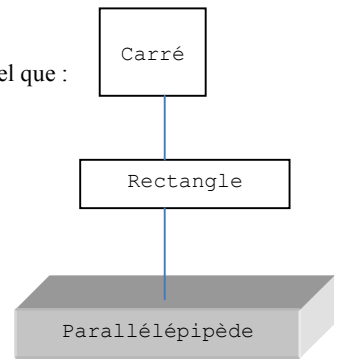
Ecrire le code qui permet de :

- (ligne 42) créer un objet y de type Livre vide,
- (ligne 43) définir le titre de l'objet y comme "Necluar Physic",
- (ligne 42) puis afficher le titre de l'objet y.

## Exercice 2 (10p)

Suivant le schéma qui montre la relation d'héritage entre les 3 classes suivantes, **Compléter le code optimal** tel que :

- La classe `Parallélépipède` possède un attribut `profondeur`, un constructeur, une méthode `volume()` et une méthode `afficher()`. A noter que  $\text{le volume} = \text{largeur} \times \text{longueur} \times \text{profondeur}$



```
class Carré
{
    public float longueur;

    public Carré(float L)
    {
        longueur = L;
    }

    public float surface()
    {
        return longueur * longueur;
    }

    public void afficher()
    {
        System.out.println("Surface Carré = "+surface()+" cm2");
    }
}-----

class Rectangle extends Carré
{
    private float largeur;

    public Rectangle(float lng, float lrg)
    {
        super(lng);
        largeur = lrg;
    }

    public float surface()
    {
        return longueur*largeur;
    }

    public void afficher()
    {
        System.out.println("Surface Rectangle = "+surface()+" cm2");
    }
}-----

class Parallelepipede extends Rectangle
{
    private float profondeur;

    public Parallelepipede(float lng, float lrg, float pfd)
    {
        super(lng,lrg);
        profondeur = pfd;
    }

    public float volume()
    {
        return super.surface()*profondeur;
    }

    public void afficher()
    {
        System.out.println("Volume = "+volume()+" cm3");
    }
}-----

public class ExamenEx2
{
    public static void main(String[] args)
    {
        Carré c = new Carré(15);
        c.afficher();
        Rectangle r = new Rectangle(12,30);
        r.afficher();
        Parallelepipede p = new Parallelepipede(4,12,9);
        p.afficher();
    }
}
```