

## Série de TP: Programmation orientée objet-Correction-

### Exercice 1

---

- 1 – Créer une classe Java nommée **Compte** qui représente un compte bancaire de visibilité public, ayant pour attributs : **public Double solde**
- 2 – Créer un constructeur ayant comme paramètre **solde**.
- 3 – Créer une méthode **deposer()** du type **void** qui gère les versements
- 4 – Créer une méthode **retirer()** du type **void** qui gère les retraits.
- 5 – Créer une méthode **afficher()** du type **void** permettant d'afficher le solde
- 6 – Donner le code complet de la classe Compte
- 7 – Créer une classe **TestCompte** permettant de tester le compte en effectuant un versement et puis un retrait

**Solution :**

#### 1 – Création de la classe Compte Java

---

```
public class Compte {  
    public Double solde;  
}
```

#### 2 – Création du constructeur avec paramètre

```
public class Compte {  
    public Double solde;  
  
    public Compte(Double solde) {  
        this.solde = solde;  
    }  
}
```

#### 3 – 4 Création des méthode déposer() et retirer()

```
public void depoter(double d){  
    this.solde += d; }  
  
public void retirer(double r){  
    this.solde -= r; }
```

#### 5 – Création de la méthode afficher()

```
public void afficher(){  
    System.out.println("Votre solde est " + this.solde + " dh "+" sauf erreur ou omission");
```

```
}
```

#### 6 – Code complet de la classe Compte

```
public class Compte {  
    public Double solde;  
  
    public Compte(Double solde) {  
        this.solde = solde;  
    }  
  
    public void deposer(double d){  
        this.solde += d;  
    }  
  
    public void retirer(double r){  
        this.solde -= r;  
    }  
  
    public void afficher(){  
        System.out.println("Votre solde est " + this.solde + " dh "+" sauf erreur ou omission");  
    }  
  
}
```

#### 7 – Création de la classe TestCompte pour tester les opérations bancaires

```
public class TestCompte {  
  
    public static void main(String[] args) {  
  
        Compte monCompte = new Compte(5000.00);  
        monCompte.deposer(3000);  
        monCompte.retirer(2000);  
        monCompte.afficher();  
  
    }  
  
}
```

***Ce qui affiche après exécution sur Eclipse :***

***Votre solde est 6000.0 dh sauf erreur ou omission***

## Exercice 2

---

- 1 – Créer une classe **Voiture** Java de visibilité public, ayant pour attributs : String marque, Double prix, de visibilité public.
- 2 – Créer un constructeur sans paramètres ( par défaut )
- 3 – Créer les getters et setters
- 4 – Créer une méthode void afficher( ) permettant d’afficher les résultats
- 5 – Donner le code final de la classe.
- 6 – Créer une classe Exécution de visibilité public permettant d’exécuter les résultats.

**Solution :**

**1 – Création de la classe Voiture :**

---

```
public class Voiture {  
    public String marque;  
    public Double prix;  
}
```

**2 – Constructeur sans paramètres**

```
public class Voiture {  
    public String marque;  
    public Double prix;  
  
    public Voiture(){  
  
    }  
}
```

**3 – Création des getters et setters**

```
public String getMarque() {  
    return marque;  
}  
  
public void setMarque(String marque) {  
    this.marque = marque;  
}  
  
public Double getPrix() {  
    return prix;  
}  
  
public void setPrix(Double prix) {  
    this.prix = prix;  
}
```

**4 – Création de la méthode void afficher( )**

```
public void afficher(){
```

```
        System.out.println("La marque de ma voiture est: "+this.marque);
        System.out.println("La prix de ma voiture est: "+this.prix);
    }
```

#### 5 – Code final de la classe

```
public class Voiture {
    public String marque;
    public Double prix;

    public Voiture(){

    }

    public String getMarque() {
        return marque;
    }

    public void setMarque(String marque) {
        this.marque = marque;
    }

    public Double getPrix() {
        return prix;
    }

    public void setPrix(Double prix) {
        this.prix = prix;
    }

    public void afficher(){
        System.out.println("La marque de ma voiture est: "+this.marque);
        System.out.println("La prix de ma voiture est: "+this.prix);
    }

}
```

#### 6 – Création de la classe Exécution

```
public class TestVoiture {

    public static void main(String[] args) {
        Voiture maVoiture = new Voiture();
        maVoiture.setMarque("Renault");
        maVoiture.setPrix(17500.00);
        maVoiture.afficher();
    }

}
```

**L'exécution sur Eclipse du code ci-dessus affiche :**

***La marque de ma voiture est : Renault***

***La prix de ma voiture est : 17500.0 dh***

### Exercice 3

---

- 1 – Créer une classe Java nommée Rectangle ayant pour attributs : Longueur et Largeur tous deux du type Double, ajouter ensuite un constructeur avec paramètres.
- 2 – Définir une méthode `surface()` du type Double permettant de calculer la surface du rectangle et une méthode `périmètre()` du type Double permettant de calculer le périmètre du rectangle.
- 3 – Créer une méthode `afficher()` du type void pour afficher les résultats. Donner ensuite le code complet de la classe.
- 4 – Créer une classe `TestRectangle` pour afficher les résultats surface et périmètre.

#### Solution :

##### 1 – Création de la classe Rectangle

---

```
public class Rectangle {  
  
    public double longueur;  
  
    public double largeur;  
  
  
  
    public Rectangle(double longueur, double largeur) {  
  
        this.longueur = longueur;  
  
        this.largeur = largeur;  
  
    }  
}
```

---

## 2 – Création des méthodes surface() et périmètre()

```
public Double surface(){
    return this.longueur*this.largeur;
}

public Double périmètre(){
    return 2*(this.largeur+this.longueur);
}
```

## Code complet de la classe

```
public class Rectangle {
    public Double largeur;
    public Double longueur;

    public Rectangle(Double largeur, Double longueur) {
        this.largeur = largeur;
        this.longueur = longueur;
    }

    public Double surface(){
        return this.largeur*this.longueur;
    }

    public Double périmètre(){
        return 2*(this.largeur+this.longueur);
    }

    public void afficher(){
        System.out.println("La surface: "+this.surface()+" Le périmètre: "+this.périmètre());
    }
}
```

## 4 – Création de la classe TestRectangle

```
public class TestRectangle {

    public static void main(String[] args) {
        Rectangle monRectangle = new Rectangle(7.0, 5.0);

        monRectangle.afficher();
    }
}
```

## Exercice 4

---

- 1 – Créer un **package Java** sous Eclipse nommé **geometry**
- 2 – Au sein du package **geometry** créer une classe Java nommée **Point** ayant pour attribut **Double Abscisse** et **double ordonnee**
- 3 – Au sein du même package créer une classe Cercle doté d'un attribut **centre** du type Point. Et des méthodes suivantes :
  - **périmètre()** du type Double permettant de calculer le périmètre du cercle.
  - **surface()** du type Double permettant de calculer la surface du cercle.
  - **testAppartenance()** du type void permettant de tester si un point appartient au cercle ou non
  - **afficher()** du type void permettant d'afficher les résultats
- 4 – Créer une classe **TestCercle** contenant une méthode **static void main()** permettant de tester les résultats

**Solution :**

### 1 et 2 – Création de la classe Point

```
package geometry;
```

```
public class Point {  
    public Double Abscisse;  
    public Double ordonnee;  
  
    public Point(Double abscisse, Double ordonnee) {  
        Abscisse = abscisse;  
        this.ordonnee = ordonnee;  
    }  
}
```

### 3 – Création de la classe Cercle

```
package geometry;
```

```
public class Cercle {  
    public Point centre;  
    public Double rayon;  
  
    public Cercle(Point centre, Double rayon) {  
        this.centre = centre;  
        this.rayon = rayon;  
    }  
}
```

```

    public Double périmètre(){
        return 2*Math.PI*rayon;
    }

    public Double surface(){
        return Math.PI*rayon*rayon;
    }

    public void testAppartenance(Point q){
        Double dx = q.Abscisse -this.centre.Abscisse;
        Double dy = q.ordonnee - this.centre.ordonnee;

        Double distance = Math.sqrt(dx*dx+dy*dy);

        if(distance.doubleValue() == this.rayon.doubleValue()){
            System.out.println("Le point appartient");
        }else{
            System.out.println("Le point n'appartient pas");
        }
    }

    public void afficher(){
        System.out.println("Le centre du cercle est le point ayant pour abscisse = " +
            this.centre.Abscisse + " ayant pour ordonnée = " +
this.centre.ordonnee);
    }

}

```

#### 4 – Création de la classe TestCercle

```

package geometry;

public class TestCercle {

    public static void main(String[] args) {
        Point centre = new Point(0.0,0.0);
        Cercle monCercle = new Cercle(centre, 3.0);

        Point M = new Point(3.0, 0.0);
        monCercle.afficher();
        monCercle.testAppartenance(M);
    }

}

```

**Ce qui affiche après exécution sur Eclipse :**

**Le rayon du cercle est 3.0**

**Le centre du cercle est le point ayant pour abscisse = 0.0 ayant pour ordonnée = 0.0**

**Le point choisi appartient au cercle**