

COMPT RENDU TP-3 POO -JAVA-

MANIPULATION DES CLASSES
ET DES OBJETS

UTILISATION DE MÉTHODES
STATIQUES

Encadrer par

Prof : M.MOUKHAFI

Préparer par

Zakaria El Omari

Informatique S5

SOMMAIRE

- 01** Exercice 1 : Articles à vendre
- 02** Exercice 2 : Du polymorphisme dans une facture
- 03** Exercice 3 : L'héritage chez les animaux

EXERCICE 1 : ARTICLES À VENDRE

EXERCICE 1 : ARTICLES À VENDRE

```
package ens.mit.article.ElOmariZakaria;
```

```
public abstract class Article {  
    private String reference;  
    private String nom;  
    private String marque;  
    private double prixUnitaire;
```

```
    public Article(String reference, String nom, String marque, double  
prixUnitaire) {  
        this.reference = reference;  
        this.nom = nom;  
        this.marque = marque;  
        this.prixUnitaire = prixUnitaire;  
    }
```

```
    public abstract double calculerPrix();
```

```
    public String getReference() {  
        return reference;  
    }
```

```
    public String getNom() {  
        return nom;  
    }
```

```
    public String getMarque() {  
        return marque;  
    }
```

```
    public double getPrixUnitaire() {  
        return prixUnitaire;  
    }
```

```
}
```

EXERCICE 1 : ARTICLES À VENDRE

```
package ens.mit.article.ElOmariZakaria;

public class Stylo extends Article {

    private String couleur;

    public Stylo(String reference, String nom, String marque, double prixUnitaire,
String couleur) {
        super(reference, nom, marque, prixUnitaire);
        this.couleur = couleur;
    }

    @Override
    public double calculerPrix() {
        return getPrixUnitaire();
    }

    public String getCouleur() {
        return couleur;
    }
}
```

```
package ens.mit.article.ElOmariZakaria;

public class RamettePapier extends Article {

    private int grammage;

    public RamettePapier(String reference, String nom, String marque, double
prixUnitaire, int grammage) {
        super(reference, nom, marque, prixUnitaire);
        this.grammage = grammage;
    }

    @Override
    public double calculerPrix() {
        return getPrixUnitaire();
    }

    public int getGrammage() {
        return grammage;
    }
}
```

EXERCICE 1 : ARTICLES À VENDRE

```
package ens.mit.article.ElOmarizakaria;
```

```
public class Lot {
    private String reference;
    private String marque;
    private Article article;
    private int nombreArticles;
    private double pourcentageReduction;

    public Lot(String reference, Article article, int nombreArticles, double
pourcentageReduction) {
        this.reference = reference;
        this.setMarque(article.getMarque());
        this.article = article;
        this.nombreArticles = nombreArticles;
        this.pourcentageReduction = pourcentageReduction;
    }

    public double calculerPrix() {
        return article.calculerPrix() * nombreArticles * (100 - pourcentageReduction) / 100;
    }

    @Override
    public String toString() {
        String nomLot = "Lot de " + nombreArticles + " " + article.getNom();
        return nomLot;
    }

    public String getReference() {
        return reference;
    }
    public Article getArticle() {
        return article;
    }
    public int getNombreArticles() {
        return nombreArticles;
    }
    public double getPourcentageReduction() {
        return pourcentageReduction;
    }
    public String getMarque() {
        return marque;
    }
    public void setMarque(String marque) {
        this.marque = marque;
    }
}
```

EXERCICE 1 : ARTICLES À VENDRE

```
package ens.mit.ElOmariZakaria.facture;

import ens.mit.article.ElOmariZakaria.Lot;

public class Facture {
    private static int compteurFactures = 0;
    private int numero;
    private String date;
    private String nomClient;
    private Lot[] lots = new Lot[20];
    private int nombreLots = 0;

    public Facture(String nomClient) {
        this.numero = ++compteurFactures;
        this.date = getDate();
        this.nomClient = nomClient;
    }

    public void ajouterLot(Lot lot) {
        lots[nombreLots] = lot;
        nombreLots++;
    }

    public double calculerPrixTotal() {
        double prixTotal = 0;
        for (int i = 0; i < nombreLots; i++) {
            prixTotal += lots[i].calculerPrix();
        }
        return prixTotal;
    }

    public void afficherFacture() {
        System.out.println("Facture #" + numero);
        System.out.println("Date: " + date);
        System.out.println("Client: " + nomClient);
        System.out.println("Détails de la facture:");

        for (int i = 0; i < nombreLots; i++) {
            System.out.println("- " + lots[i].toString() + " à " + lots[i].calculerPrix() + " DH");
        }

        System.out.println("Prix total : " + calculerPrixTotal() + " DH\n");
    }
}
```

EXERCICE 1 : ARTICLES À VENDRE - SUITE

```
public int getNumero() {  
    return numero;  
}  
  
public String getDate() {  
    return "01/01/2023";  
}  
  
public String getNomClient() {  
    return nomClient;  
}  
}
```


EXERCICE 1 : ARTICLES À VENDRE

```
package ens.mit.ElOmariZakaria.facture;

import ens.mit.article.ElOmariZakaria.Article;
import ens.mit.article.ElOmariZakaria.Lot;
import ens.mit.article.ElOmariZakaria.RamettePapier;
import ens.mit.article.ElOmariZakaria.Stylo;

public class Test {
    public static void main(String[] args) {

        ////////// Client 1 : //////////////////////
        Article stylo = new Stylo("REF001", "Stylo Noir", "WaterTruc", 10.0, "Noir");
        Article ramette = new RamettePapier("REF002", "Ramette A4", "PaperCo", 15.0,
80);

        Lot lotStylos1 = new Lot("REF003", stylo, 5, 20); /// reduction 20%
        Lot lotStylos2 = new Lot("REF005", stylo, 5, 20); /// reduction 20%
        Lot lotRamettepapier = new Lot("REF006", ramette, 15, 20); /// reduction 20%

        Facture facture1 = new Facture("Client1");
        facture1.ajouterLot(lotStylos1);
        facture1.ajouterLot(lotStylos2);
        facture1.ajouterLot(lotRamettepapier);

        facture1.afficherFacture();

        ////////// Client 1 : //////////////////////
        Lot lotStylos3 = new Lot("REF007", stylo, 3, 20); /// reduction 20%
        Lot lotRamettepapier2 = new Lot("REF008", ramette, 2, 20); /// reduction 20%

        Facture facture2 = new Facture("Client2");
        facture2.ajouterLot(lotStylos3);
        facture2.ajouterLot(lotRamettepapier2);

        facture2.afficherFacture();
    }
}
```

EXERCICE 2 : DU POLYMORPHISME DANS UNE FACTURE

EXERCICE 2 : DU POLYMORPHISME DANS UNE FACTURE

```
package ens.mit.ElOmariZakaria.facture;

import ens.mit.article.ElOmariZakaria.Lot;

public class Facture {
    private static int compteurFactures = 0;
    private int numero;
    private String date;
    private String nomClient;
    private Lot[] lots = new Lot[20];
    private int nombreLots = 0;

    public Facture(String nomClient) {
        this.numero = ++compteurFactures;
        this.date = getDate();
        this.nomClient = nomClient;
    }

    public void ajouterLot(Lot lot) {
        lots[nombreLots] = lot;
        nombreLots++;
    }

    public double calculerPrixTotal() {
        double prixTotal = 0;
        for (int i = 0; i < nombreLots; i++) {
            prixTotal += lots[i].calculerPrix();
        }
        return prixTotal;
    }

    public void afficherFacture() {
        System.out.println("Facture #" + numero);
        System.out.println("Date: " + date);
        System.out.println("Client: " + nomClient);
        System.out.println("Détails de la facture:");

        for (int i = 0; i < nombreLots; i++) {
            System.out.println("- " + lots[i].toString() + " à " + lots[i].calculerPrix() + " DH");
        }

        System.out.println("Prix total : " + calculerPrixTotal() + " DH\n");
    }
}
```

EXERCICE 2 : DU POLYMORPHISME DANS UNE FACTURE - SUITE

```
public int getNumero() {  
    return numero;  
}  
  
public String getDate() {  
    return "01/01/2023";  
}  
  
public String getNomClient() {  
    return nomClient;  
}  
}
```

EXERCICE 2 : DU POLYMORPHISME DANS UNE FACTURE

```
package ens.mit.ElOmariZakaria.facture;

import ens.mit.article.ElOmariZakaria.Article;
import ens.mit.article.ElOmariZakaria.Lot;
import ens.mit.article.ElOmariZakaria.RamettePapier;
import ens.mit.article.ElOmariZakaria.Stylo;

public class Test {
    public static void main(String[] args) {

        ////////// Client 1 : ///////////////////
        Article stylo = new Stylo("REF001", "Stylo Noir", "WaterTruc", 10.0, "Noir");
        Article ramette = new RamettePapier("REF002", "Ramette A4", "PaperCo", 15.0,
80);

        Lot lotStylos1 = new Lot("REF003", stylo, 5, 20); /// reduction 20%
        Lot lotStylos2 = new Lot("REF005", stylo, 5, 20); /// reduction 20%
        Lot lotRamettepapier = new Lot("REF006", ramette, 15, 20); /// reduction 20%

        Facture facture1 = new Facture("Client1");
        facture1.ajouterLot(lotStylos1);
        facture1.ajouterLot(lotStylos2);
        facture1.ajouterLot(lotRamettepapier);

        facture1.afficherFacture();

        ////////// Client 1 : ///////////////////
        Lot lotStylos3 = new Lot("REF007", stylo, 3, 20); /// reduction 20%
        Lot lotRamettepapier2 = new Lot("REF008", ramette, 2, 20); /// reduction 20%

        Facture facture2 = new Facture("Client2");
        facture2.ajouterLot(lotStylos3);
        facture2.ajouterLot(lotRamettepapier2);

        facture2.afficherFacture();
    }
}
```

EXERCICE 2 : DU POLYMORPHISME DANS UNE FACTURE

Exercice : 2 exécution

```
Console ×
<terminated> Test [Java Application] /snap/eclipse/73/plugins/org.eclip
Facture #1
Date: 01/01/2023
Client: Client1
Détails de la facture:
- Lot de 5 Stylo Noir à 40.0 DH
- Lot de 5 Stylo Noir à 40.0 DH
- Lot de 15 Ramette A4 à 180.0 DH
Prix total : 260.0 DH

Facture #2
Date: 01/01/2023
Client: Client2
Détails de la facture:
- Lot de 3 Stylo Noir à 24.0 DH
- Lot de 2 Ramette A4 à 24.0 DH
Prix total : 48.0 DH
```

EXERCICE 3 : L'HÉRITAGE CHEZ LES ANIMAUX

EXERCICE 3 : L'HÉRITAGE CHEZ LES ANIMAUX

```
package ex2;
```

```
public class Animal {
    private String nom;

    public Animal(String nom) {
        this.nom = nom;
    }

    public Animal() {
        this.nom = "Sans nom";
    }

    public String getType() {
        return "Je suis un animal" + (nom.isEmpty() ? "." : " de nom " + nom + ".");
    }
}
```

```
package ex2;
```

```
public class Mammifere extends Animal {
    public Mammifere(String nom) {
        super(nom);
    }

    public Mammifere() {
        super();
    }

    @Override
    public String getType() {
        return super.getType() + " Je suis un mammifère.";
    }
}
```


EXERCICE 3 : L'HÉRITAGE CHEZ LES ANIMAUX

```
package ex2;
```

```
public class Chien extends Mammifere {  
    public Chien(String nom) {  
        super(nom);  
    }  
  
    @Override  
    public String getType() {  
        return super.getType() + " Je suis un chien.";  
    }  
}
```

```
package ex2;
```

```
public class Lion extends Mammifere {  
    public Lion(String nom) {  
        super(nom);  
    }  
  
    public Lion() {  
        super();  
    }  
  
    @Override  
    public String getType() {  
        return super.getType() + " Je suis un lion.";  
    }  
}
```

EXERCICE 3 : L'HÉRITAGE CHEZ LES ANIMAUX

```
package ex2;

public class TestAnimal {
    public static void main(String[] args) {
        Animal[] animaux = new Animal[5];
        animaux[0] = new Animal("Truc");
        animaux[1] = new Animal();
        animaux[2] = new Chien("Médor");
        animaux[3] = new Lion();
        animaux[4] = new Lion("Robert");

        for (int i = 0; i < animaux.length; i++) {
            System.out.println(animaux[i].getType());
        }
    }
}
```

EXERCICE 3 : L'HÉRITAGE CHEZ LES ANIMAUX

Exercice : 3 exécution

Console ×

```
<terminated> TestAnimal [Java Application] /snap/eclipse/73/plugins/org.eclipse.justj.openjdk  
Je suis un animal de nom Truc.  
Je suis un animal de nom Sans nom.  
Je suis un animal de nom Médor. Je suis un mammifère. Je suis un chien.  
Je suis un animal de nom Sans nom. Je suis un mammifère. Je suis un lion.  
Je suis un animal de nom Robert. Je suis un mammifère. Je suis un lion.
```