

La programmation orienté objet

I. Introduction

dans la programmation structurée PS, un programme se présente comme une composition de deux parties :

- Les Fichiers de données
- Les action (procédures/fonctions) qui agissent sur les objets de données.

Les insuffisances de la PS :

- manque de lisibilité et de visibilité des codes
- difficulté de la réutilisabilité des codes
- difficulté de maintenance des codes

La POO va remédier aux problèmes énoncés, la POO tente de **modéliser** les objets du monde réel tels qu'ils sont perçus sous forme d'un état et d'un comportement.

Etat : ensemble de propriétés, attributs, caractéristiques qui définissent l'objet intrinsèquement.

Comportement : ensemble des méthodes (procédures et fonctions) qui agissent sur l'Etat d'un objet et le font évoluer.

```
Exemple :
    Etat : nom, prenom, age, taille, ...
    Homme
    Comportement : naitre, grandir, voyager ... mourrir
    Etat: nom, prenom, numero, solde, ...
    Compte Bancaire
    Comportement : ouvrir, déposer, retirer, ... fermer
```

En conclusion, un programme en Orienté se présente comme un ensemble d'objets qui peuvent communiquer entre eux.

schéma

II - Les concepts de base de la POO

1. Les Classes et les objets

Une **classe** est une **famille**, un **modèle**, un **superType** qui possède un nom et est composé d'un état et d'un comportement.

Un **objet** est une **réalisation** de la classe dans le monde réel, une **instance** de la classe, une **sorte**, une **déclinaison** dans le monde réel.

schéma

Syntax de déclaration :

```
Algo:
    classe nomClasse
        //attributs
        //méthodes
    fin nomClasse
```

Java et PHP

```
classe nomClasse
{
    //attributs
    //Les methodes
}
```

Création d'un objet : instanciation de la classe.

```
Algo:
  nomObjet <- new nomClasse ()
```

PHP :

```
$nomObjet = new nomClasse();
```

Java

```
nomObjet = new nomClasse();
```

2. La visibilité

L'accès à un membre de la classe (attribut ou méthode) peut se faire selon trois types de visibilité : privé (private), publique(public) et protégé(protected).

Privé : tout membre déclaré en privé est accessible exclusivement dans sa classe.

public : Tout membre déclaré en public est accessible dans sa classe et à l'extérieur de celle-ci.

protégé : Tout membre déclaré en protégé est accessible dans sa classe et dans toutes les classes héritières (filles), principe d'héritage

principe d'encapsulation : règle générale : tous les attributs sont déclarés en PRIVE, sauf exception, toutes les méthodes sont déclarées en PUBLIC sauf exception

syntax : déclarer une classe Date (jour, mois, année, méthode : saisir () et afficher ())

Algo :

```
Algo:
  classe Date
    privé :
      jour, mois, année : entier
    public :
      procédure a saisir ()
      début
        ...
      fin saisir
      procédure afficher ()
      début
        ...
      fin afficher()
  fin Date()
```

Java :

```
class Date
{
  private int jour, mois, annee ;
  public void saisir()[...];
  public void afficher()[...];
}
```

PHP :

```
class Date
{
  private $jour, $mois, $annee;
  public function saisir()[...];
  public function afficher()[...];
}
```

Comment accéder aux attributs privés d'une classe à partir d'une autre classe ?

On crée dans la classe deux méthodes pour chaque attribut privé :

- un getter : une méthode qui permet de retourner la valeur de l'attribut appelé par défaut getNomAttribut()
- un setter : méthode procédure qui permet de modifier la valeur de l'attribut appelé par défaut setNomAttribut()

Exemple en Java pour l'attribut jour :

```
int getJour ();
{
    return jour;
}
void setJour (int j)
{
    jour = j;
}
```

3. La référence this

Chaque objet incorpore en son sein une référence appelée par défaut `this`, qui permet l'auto-référencement (le moi). `This` accède à l'ensemble des attributs et méthodes de la classe.

Il lève l'ambiguïté de noms entre les attributs de la classe et les arguments des méthodes. Il indique l'objet courant.

Exemple : reprenons l'exemple du setter du jour en Java, en renommant l'argument de la méthode avec le même nom que l'attribut :

```
void setJour (int jour)
{
    this.jour = jour; //levé de l'ambiguïté avec this.
}
```

syntax en PHP :

```
public function setJour ($jour)
{
    $this->jour = $jour;
}
```

4. Les constructeurs et les destructeurs

Les constructeurs sont des méthodes particulières qui s'exécutent dès l'instanciation d'une classe automatiquement (sans appel). Ils permettent l'initialisation des attributs de la classe.

En Java et Algo, les constructeurs ont le même nom que la classe. En PHP, les constructeurs sont nommés `__construct()` ; (à partir de PHP5)

En Java, il n'y a pas de destructeurs, (Java dispose d'un programme ramasse-miettes qui permet de récupérer les espaces mémoires de objets non utilisés).

En PHP, les destructeurs sont nommés `__destroy()`, ils s'exécutent à la fin de vie des objets automatiquement.

----- Fin du Cours -----

Exercice Compte Java

Compte.java

```
import java.util.Scanner;

public class Compte {

    // attributs
    private int numero;
    private String nom, prenom;
    private float solde;

    // constructeur
    public Compte()
    {
        this.numero = 0;
        this.nom = '';
        this.prenom = '';
        this.solde = 80;
    }

    //methodes
    public void ouvrir(){
        Scanner sc = new Scanner (System.in); //Clavier
        System.out = ecran;
        System.out.println("Donner le numero du compte : ");
    }
}
```

```

        this.numero = sc.nextInt();
        System.out.println("Donner le nom du client : ");
        this.nom = sc.next();
        System.out.println("Donner le prenom du client : ");
        this.prenom = sc.next();
    }
    public void afficher() {
        System.out.println("Numero du compte : " + this.numero);
        System.out.println("Nom du client : " + this.nom);
        System.out.println("Prénom du client : " + this.prenom);
        System.out.println("Solde du compte : " + this.solde + "Euros \n");
    }
    public void depoter() {
        float somme;
        Scanner sc = new Scanner (System.in);
        System.out.println("Donner la somme à déposer : ");
        somme = sc.nextFloat();
        this.solde = this.solde + somme;
        System.out.println("Votre nouveau solde est de : " + this.solde + "euro");
    }
    public void retirer() {
        float somme;
        Scanner sc = new Scanner (System.in);
        System.out.println("Donner la somme à retirer : ");
        somme = sc.nextFloat();
        if(somme > this.solde){
            System.out.println("Retrait Impossible !");
        }
        else {
            this.solde = this.solde - somme;
            System.out.println("Votre nouveau solde est de : " + this.solde + "euros")
        }
    }
    public void gerer() {

    }

    // getters and setter
}

```

Exercice Stock java :

Produit.java

```

import java.util.Scanner;

public class Produit {
    // Attributs
    private String ref, des;
    private float prix;
    private int qte;

    // Constructeur
    public Produit() {
        this.ref = "";
        this.des = "";
        this.prix = 0;
        this.qte = 0;
    }

    // Methodes
    public void saisir() {
        Scanner sc = new Scanner(System.in);

        System.out.println("Donner la référence : ");
        this.ref = sc.next();

        System.out.println("Donner la désignation du produit : ");
        this.des = sc.next();
    }
}

```

```

        System.out.println("Donner le prix du produit : ");
        this.prix = sc.nextFloat();

        System.out.println("Donner la quantité en stock : ");
        this.qte = sc.nextInt();
    }

    public void afficher() {
        System.out.println("La référence du produit est : " + this.ref);
        System.out.println("La désignation du produit est : " + this.des);
        System.out.println("Le prix du produit est : " + this.prix);
        System.out.println("La quantité en stock est : " + this.qte);
    }

    public void vendre() {
        Scanner sc = new Scanner(System.in);
        int qteavendre;
        System.out.println("Quel est la quantité à vendre ? : ");
        qteavendre = sc.nextInt();
        if(qteavendre <= this.qte) {
            this.qte = this.qte - qteavendre;
            System.out.println("Vente réalisée, quantité en stock : " + this.qte);
        }
        else {
            System.out.println("Vente Impossible");
        }
    }

    public void approvisionner() {
        Scanner sc = new Scanner(System.in);
        int qteachetes;
        System.out.println("Donner la quantité achetée : ");
        qteachetes = sc.nextInt();
        this.qte = this.qte + qteachetes;
        System.out.println("La Quantité en stock est de : " + this.qte);
    }

    public float totalCA() {
        float total;
        total = this.prix * this.qte;
        return total;
    }

    public void gerer() {
        Scanner sc = new Scanner(System.in);
        int choix;
        do {
            System.out.println("_____ Menu Stock _____");
            System.out.println("|");
            System.out.println("|      1. Saisir Produit      |");
            System.out.println("|");
            System.out.println("|      2. Afficher Produit   |");
            System.out.println("|");
            System.out.println("|      3. Vender              |");
            System.out.println("|");
            System.out.println("|      4. Approvisionner     |");
            System.out.println("|");
            System.out.println("|      5. Total CA           |");
            System.out.println("|");
            System.out.println("|      0. Quitter            |");
            System.out.println("_____");
            System.out.println("Quel est votre choix : ");
            choix = sc.nextInt();
            switch(choix) {
                case 1 : this.saisir();
                break;
                case 2 : this.afficher();
                break;
            }
        } while (choix != 0);
    }

```

```

        case 3 : this.vendre();
        break;
        case 4 : this.approvisionner();
        break;
        case 5 : System.out.println("Total CA :"+ this.totalCA());
        break;
        case 0 : System.out.println("Fin");
        break;
        default: System.out.println("erreur choix")
        ;break;
    }
}while(choix !=0);
}

//Getters and Setters

public String getRef() {
    return ref;
}

public void setRef(String ref) {
    this.ref = ref;
}

public String getDes() {
    return des;
}

public void setDes(String des) {
    this.des = des;
}

public float getPrix() {
    return prix;
}

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

public int getQte() {
    return qte;
}

public void setQte(int qte) {
    this.qte = qte;
}
}
}

```

Stock.java

```

public class Stock {

    public static void main (String args[]) {
        // instruction de la classe Produit : creation d'un objet de cette classe
        Produit unProduit = new Produit();

        // Appel de la méthode gerer :
        unProduit.gerer();
    }
}

```

Exercice Nombre

Nombre.java

```

import java.util.Scanner;

public class Nombre {
    private int nb;

    public Nombre() {
        this.nb = 0;
    }

    public void saisir() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Donner Votre nombre : ");
        this.nb = sc.nextInt();
    }

    public void afficher() {
        System.out.println("Votre nombre est : " + this.nb);
    }

    public void pairImpair() {
        if(this.nb % 2 == 0) {
            System.out.println("Votre nombre est pair");
        }
        else {
            System.out.println("Votre nombre est impair");
        }
    }

    public void mesDiviseurs() {
        int i = 1;
        System.out.println("voici les diviseurs du nombre " + this.nb);
        while(i <= this.nb) {
            if(this.nb % i == 0) {
                System.out.println(i + "est un diviseur de " + this.nb);
            }
            i++;
        }
    }

    public void tableMultiplication() {
        Scanner sc= new Scanner(System.in);
        int limite;
        System.out.println("Donner la limite de la table : ");
        limite = sc.nextInt();
        for(int i= 1; i<=limite; i++) {
            System.out.println(i + "*" + this.nb + "=" + (i*this.nb));
        }
    }

    public void factoriel() {
        int fact = 1;
        for(int i = 1; i<=this.nb; i++) {
            fact = fact*i;
        }
        System.out.println("Le Factoriel : " + fact);
    }

    public void puissance() {
        int exposant;
        Scanner sc = new Scanner(System.in);
        System.out.println("Donner l'exposant");
        exposant = sc.nextInt();
        int p = 1;
        for(int i =1; i<=exposant; i++) {
            p = p * this.nb;
        }
        System.out.println("La puissance est de : " + p);
    }

    public void gerer() {
        Scanner sc = new Scanner(System.in);
    }

```

```

        int choix;
        do {
            System.out.println("          Menu Stock          ");
            System.out.println("|");
            System.out.println("|      1. Saisir Nombre      |");
            System.out.println("|");
            System.out.println("|      2. Afficher Nombre   |");
            System.out.println("|");
            System.out.println("|      3. Diviseurs         |");
            System.out.println("|");
            System.out.println("|      4. Table Multiplication |");
            System.out.println("|");
            System.out.println("|      5. Factoriel         |");
            System.out.println("|");
            System.out.println("|      5. Puissance         |");
            System.out.println("|");
            System.out.println("|      0. Quitter          |");
            System.out.println("          ");
            System.out.println("Quel est votre choix : ");
            choix = sc.nextInt();
            switch(choix) {
                case 1 : this.saisir();
                break;
                case 2 : this.afficher();
                break;
                case 3 : this.mesDiviseurs();
                break;
                case 4 : this.tableMultiplication();
                break;
                case 5 : this.factoriel();
                break;
                case 6 : this.puissance();
                break;
                case 0 : System.out.println("Fin");
                break;
                default: System.out.println("erreur choix")
                ;break;
            }
        }while(choix !=0);
    }

    //Getters and Setters

    public int getNb() {
        return nb;
    }

    public void setNb(int nb) {
        this.nb = nb;
    }
}

```

Gestion.java

```

public class Gestion {
    public static void main (String args[]) {
        // instruction de la classe Produit : creation d'un objet de cette classe
        Nombre unNombre = new Nombre();

        // Appel de la méthode gerer :
        unNombre.gerer();
    }
}

```


Exercice Tableau

Tableau.java

```
import java.util.Arrays;
import java.util.Scanner;

public class Tableau {
    private float tab[];

    public Tableau (int taille) {
        // initialisation du tableau
        this.tab = new float [taille];
    }

    public void remplir() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Remplissage du tableau : ");
        for(int i = 0; i < this.tab.length; i++) {
            System.out.println("Donner les elements " + i + " : ");
            this.tab[i] = sc.nextFloat();
        }
    }

    public void afficher() {
        System.out.println("Tableau : " + Arrays.toString(tab));
    }

    public float moyenne() {
        int somme = 0;
        for(int i = 0; i < tab.length; i++){
            somme += tab[i];
        }
        float moy = (float) somme / tab.length;
        return moy;
    }

    public void min_max() {
        float maxVal = this.tab[0];
        float minVal = this.tab[0];
        for(int i = 0; i < tab.length; i++){
            if(tab[i] < maxVal)
                maxVal = tab[i];
            if(tab[i] > minVal)
                minVal = tab[i];
        }

        System.out.println("\nValeur minimale = " + maxVal);
        System.out.println("\nValeur maximale = " + minVal);
    }

    public void rechercher() {
        Scanner sc = new Scanner(System.in);
        float valeur;
        int cpt = 0;
        System.out.println("Entrer la valeur rechercher : ");
        valeur = sc.nextFloat();
        for(int i=0; i<this.tab.length; i++) {
            if(this.tab[i] == valeur)
            {
                cpt ++;
            }
        }
        System.out.println("La valeur est présente "+ cpt + " fois");
    }

    public void trier() {
        int indicemin;
        float min;
    }
}
```

```

        for(int i = 0; i<this.tab.length - 1; i++) {
            min = this.tab[i];
            indicemin = i;
            for(int j=i+1; j<this.tab.length; j++) {
                if(this.tab[j]<min) {
                    min = this.tab[j];
                    indicemin = j;
                }
            }
            this.tab[indicemin] = this.tab[i];
            this.tab[i] = min;
        }
        System.out.println("Voici le tableau trier en ordre croissant : ");
        this.afficher();
    }

    public void gerer() {
        Scanner sc = new Scanner(System.in);
        int choix;
        do {
            System.out.println("_____ Menu Stock _____");
            System.out.println("|");
            System.out.println("|      1. Remplir      |");
            System.out.println("|");
            System.out.println("|      2. Afficher    |");
            System.out.println("|");
            System.out.println("|      3. Moyenne     |");
            System.out.println("|");
            System.out.println("|      4. Min et Max  |");
            System.out.println("|");
            System.out.println("|      5. Rechercher  |");
            System.out.println("|");
            System.out.println("|      6. Trier       |");
            System.out.println("|");
            System.out.println("|      0. Quitter     |");
            System.out.println("_____");
            System.out.println("Quel est votre choix : ");
            choix = sc.nextInt();
            switch(choix) {
                case 1 : this.remplir();
                    break;
                case 2 : this.afficher();
                    break;
                case 3 : System.out.println("moyenne : "+ this.moyenne());
                    break;
                case 4 : this.min_max();
                    break;
                case 5 : this.rechercher();
                    break;
                case 6 : this.trier();
                    break;
                case 0 : System.out.println("Fin");
                    break;
                default: System.out.println("erreur choix")
                    ;break;
            }
        }while(choix !=0);
    }

    // Getters and Setters
    public float[] getTab() {
        return tab;
    }

    public void setTab(float[] tab) {
        this.tab = tab;
    }
}

```

```

public class Gestion {

    public static void main(String[] args) {
        Tableau unTab = new Tableau(10);
        unTab.gerer();
    }
}

```

Phrase

Phrase.java

```

import java.util.Scanner;

public class Phrase {

    private String phrase;

    public Phrase() {

    }

    public void saisir() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Saisir votre phrase : ");
        this.phrase = sc.next();
    }

    public void afficher() {
        System.out.println("votre phrase est : \n" + this.phrase);
    }

    public int occurrence(char lettre) {

        int count = 0;
        for(int i = 0; i<this.phrase.length(); i++) {
            if(this.phrase.charAt(i) == lettre ) {
                count ++;
            }
        }
        return count;
    }

    public String miroir() {
        String m = "";
        for(int i = this.phrase.length(); i>=0; i--) {
            m = m + this.phrase.charAt(i);
        }
        return m;
    }

    public void palindrome() {
        String pal = this.miroir();
        if(pal.equalsIgnoreCase(phrase)) {
            System.out.println("Cette phrase est un palindrome");
        }
        else {
            System.out.println("Cette phrase n'est pas un palindrome");
        }
    }

    public String crypter(int cle) {
        String Crypt = "";
        for(int i=0; i<this.phrase.length(); i++) {
            Crypt = Crypt + (char)(phrase.charAt(i) + cle);
        }
    }
}

```

```

        return Crypt;
    }

    public String decrypter(int cle) {
        String decrypt = "";
        for(int i=0; i<this.phrase.length(); i++) {
            decrypt = decrypt + (char)(phrase.charAt(i) - cle);
        }
        return decrypt;
    }

    public void gerer() {
        Scanner sc = new Scanner(System.in);
        int choix;
        do {
            System.out.println("_____ Menu Stock _____");
            System.out.println("|");
            System.out.println("|      1. Saisir      |");
            System.out.println("|");
            System.out.println("|      2. Afficher   |");
            System.out.println("|");
            System.out.println("|      3. Occurence  |");
            System.out.println("|");
            System.out.println("|      4. Palindrome |");
            System.out.println("|");
            System.out.println("|      5. Crypter    |");
            System.out.println("|");
            System.out.println("|      6. decrypter  |");
            System.out.println("|");
            System.out.println("|      0. Quitter    |");
            System.out.println("_____");
            System.out.println("Quel est votre choix : ");
            choix = sc.nextInt();
            switch(choix) {
                case 1 : this.saisir();
                    break;
                case 2 : this.afficher();
                    break;
                case 3 : {
                    char lettre;
                    System.out.println("Lettre à rechercher : ");
                    lettre = sc.next().charAt(0);
                    System.out.println("Occurence : " + this.occurence(lettre));
                }
                    break;
                case 4 : this.palindrome();
                    break;
                case 5 : {
                    int cle;
                    System.out.println("Quel est la cle de cryptage : ");
                    cle = sc.nextInt();
                    System.out.println("Chaine crypter : \n" + this.crypter(cle));
                }
                    break;
                case 6 : {
                    int cle;
                    System.out.println("Quel est la cle de decryptage : ");
                    cle = sc.nextInt();
                    System.out.println("Chaine decrypter : \n" + this.decrypter(cle));
                }
                    break;
                case 0 : System.out.println("Fin");
                    break;
                default: System.out.println("erreur choix");
                    break;
            }
        }while(choix !=0);
    }
    //getters and setters

```

```
public String getPhrase() {  
    return phrase;  
}  
  
public void setPhrase(String phrase) {  
    this.phrase = phrase;  
}  
}
```

GestionPhrase.java

```
public class GestionPhrase {  
  
    public static void main(String[] args) {  
        Phrase unePhrase = new Phrase();  
        unePhrase.gerer();  
    }  
}
```