

CLASSE ET OBJETS

Exercice1

Définissez la classe *Cercle* :

```
public class Cercle {
float r;

// constructeurs

public Cercle(){// par défaut
System.out.println('création par défaut') ;}

public Cercle(float r){//avec paramètres
    this.r=r;
System.out.println('création et initialisation `') ;
}

public float Perimetre(){
return 2*(float) Math.PI*r;
}
public float Surface(){
return (float) Math.PI*(r*r);
}

public static void main(String[] args) {
Cercle C1=new Cercle(); C1.r=5;
Cercle C2=new Cercle(3);
float S1= C1.Surface(5);
float S2= C2.Surface(3);
float Minimum= Math.min(S1, S2);
float P1= C1.Perimetre(5);
float P2= C2.Perimetre(3);
float Maximum= Math.max(P1, P2);
System.out.println("minimum="+Minimum+"Maximum="+Maximum);
}
}
```

Exercice2

```
public class Personne {
String nom;
String prenom;
String fonction;

public Personne{
}
public Personne(String Nom,String Prenom ){
this.nom=Nom;
this.prenom=Prenom;
}

public Personne(String Nom,String Prenom,String Fonction ){
this.nom=Nom;
this.prenom=Prenom;
this.fonction=Fonction;
}
Public void afficher_nomcomplet(){
S.O.P('`le nom complet : `'+ Nom+ ` `'+ Prenom) ;
}
}
```

```
Public void afficher2 () {  
S.O.P(`le nom complet : `+ Nom+ ` `+ Prenom+ ` la fonction :'+ Fonction) ;  
}
```

La manipulation :

```
public static void main(String [] args){  
Personne P1= new Personne();  
P1.nom="ahmed";  
P1.prenom="said";  
P1.fonction="administratif";  
public static void main () {  
Personne P2= new Personne("benazi","ahmed");  
Personne P3= new Personne("saadi","ali", "directeur");  
P2.nom="saoudi";  
P2.fonction="etudiant";  
P1.afficher_nomcomplet();  
p2.afficher2();  
}  
}
```

Exercice 3

```
public class CompteBancaire {  
float solde=0;  
public void deposer(float montant){  
solde+=montant;  
}  
public void retirer(float montant){  
if (solde <= montant) solde-=montant;  
else System.out.println("erreur");  
}  
public void afficher(){  
System.out.println("solde="+solde);  
}  
}
```

La classe *Manipulation*

```
public class ManipulationSolde{  
public static void main(String[] args) {  
CompteEnBanque comptel = new CompteBanque();  
CompteEnBanque comptee = new CompteBanque();  
Comptel.deposer(5000) ;  
Comptel.retirer(1000) ;  
Comptee.deposer(5000) ;  
System.out.println(comptel.afficher());  
System.out.println(comptee.afficher());  
}  
}
```

ENCAPSULATION & HÉRITAGE

Exercice 1

Modificateur	Classe	paquetage	Sous-classe	Autre classe
Private	visible			
Aucun	visible	visible		
Protected	visible	visible	visible	
Public	visible	visible	visible	visible

1. Dans une classe si on a 7 attributs privés, on a besoin de 7 méthodes accesseurs et 7 méthodes modificateurs (pour chaque attribut on définit une méthode accesseur *get* et une méthode modificateur *set*)
2. Oui, on peut définir plus qu'une méthode **constructeur** dans une même classe, si on veut implémenter plusieurs façon d'initialisation pour l'objet. Mais cette contrainte est rarement utilisée.
3. La surcharge des méthodes signifie qu'on peut définir au sein de même classe des méthodes ayant **le même nom** mais avec un nombre ou type de paramètre différent.

Exercice 02

```
public class Livre {
private int id;
private String titre;
private String auteur;
private int prix;
public static int compteur=0;

public Livre(String titre, String auteur, int prix) {
this.id = ++compteur;
this.titre = titre;
this.auteur = auteur;
this.prix = prix;
}
public int getId() {
return id;
}
public void setId(int id) {
this.id = id;
}
public String getTitre() {
return titre;
}
public void setTitre(String titre) {
this.titre = titre;
}
public String getAuteur() {
return auteur;
}
public void setAuteur(String auteur) {
this.auteur = auteur;
}
public int getPrix() {
return prix;
}
public void setPrix(int prix) {
this.prix = prix;
}
}
```

Exercice 03

```
public class encapsulation{
public int i1=10;
private int i2=20;
protected int i3=30;
private void m2(){ i2+=10; }
```

□ Deux méthodes *get* et *set*:

```
Public int getI2(){ return i2; }
```

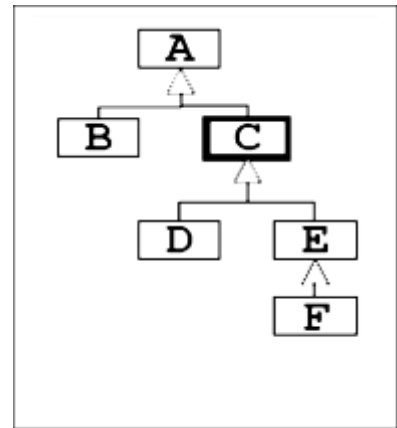
Cette méthode est un *accesseur*, elle permet de donner des valeurs de variables privée de l'extérieur de la classe.

```
public void setI2(int i2){ this.i2=i2; }
```

Exercice 04

```
Class A{ }
Class B extends A{ }
Class C extends A{ }
Class D extends C{ }
Class E extends C { }
Class F extends E{ }
```

```
A a; B b; C c; D d; E e;
a= new B();
b= new A();
e= new B();
c= new D();
d= new D();
d= new A();
e= new B();
```



Exercice 05

```
Public class Dictionnaire extends Livre{
Private int nbreTerm,nbreLang ;
}
Public class Document extends Livre{
Private int nbrePage ;
```

- Dans la classe Dictionnaire il y a 6 attributs : 4 de la classe Livre + 2
- Dans la classe Document il y a 5 attributs : 4 de la classe Livre +1

Exercice 6

```
public class Employe extends Personne{
private float salaire;
public Employe (String nom, String prenom, string fonction, float salaire) {
super(nom, prenom, Fonction);
this.salaire=saliare;
}
public void set_saliare (float salaire){
this.salaire=salaire;
}
public float get_salaire(){
return salaire ;
}
}
```

```
public class Employecadre extends Employe{
private float prime ;

public Employecadre(String nom, String prenom, string fonction, float
salaire,float prime) {
super(nom, prenom, fonction, salaire);
this.prime=prime;

public void set_prime (float prime){
this.prime=prime;
}
public float get_prime(){
return prime ;
}
}
```