Université Badji Mokhtar Annaba
Département d'informatique
2ème Année LMD
Module POO
Année 2022/2023

Série TD 1:notions de base POO

Exercice 01 :

Soit le scénario simple d'une réservation de vol :

• Mohamed désire avoir une réservation pour aller par avion à Alger, Vendredi prochain.

• Mohamed demande à Ali (qui travaille pour l'agence de voyage Hippone-Tour) d'effectuer une réservation par avion pour Alger.

• Ali demande à Omar (qui travaille pour la compagnie aérienne Air-Algerie) l'horaire.

• Omar consulte la base de données des horaires du vendredi et informe Ali.

• Ali informe à son tour Mohamed de l'horaire.

1. Quels sont les objets de ce scénario ?
2. Quelles sont les interactions entre ces différents objets?
3. Regrouper les objets par Classe d'objets.

Exercice 02 : Soit la classe *F* définie ci-dessous. f est une instance de *F*.

```
public class F {
int i;
static String s;
void imethod() {
}
static void smethod() {
}
}
```

Quelles sont les instructions invalides, parmi les instructions suivantes :

```
System.out.println(f.i);
System.out.println(f.s);
f.imethod();
f.smethod();
System.out.println(F.i);
System.out.println(F.s);
F.imethod();
F.smethod();
```

Exercice 03:

Quel est le résultat de l'exécution du code suivant :

```
public class Test {
public static void main(String[] args) {
A a1 = new A();
System. out .println(a1.i);
System. out .println(a1.j);
A a2 = new A();
System. out .println(a2.i);
System. out .println(a2.j);
}
}
class A {
int i = 1;
static int j = 1;
A() {
i++;
j++;
}
}
```

Exercice 04 : Considérer le code suivant :

```
Rectangle box1 = new Rectangle(5, 10, 20, 30);
Rectangle box2 = box1;
Rectangle box3 = new Rectangle(5, 10, 20, 30);
```

Quel serait le résultat de la comparaison

```
box1 == box2?
```

Quel serait le résultat de la comparaison

```
box1 == box3?
```

Exercice 05 : Soit la définition des deux méthodes suivantes :

```
Public static double m(double x,double y)
public static double m(int x, double y)
```

Quelle méthode sera invoquée dans ce qui suit :

```
1.  double z = m(4, 5);
2.  double z = m(4, 5.4);
```

```
3.  double z = m(4.5, 5.4);
```

Exercice 06: Soit la classe :

```
class Fleur {
private
int petales;
int tige;
int pedoncule;
int etamines;
protected
static int pollen;
public
Fleur();
Fleur( int p, int t)
void setPetales( int p);
int getEtamines();
}
```

1. Identifier les variables de classe et les variables d'instance dans la classe Fleur, et donner le nombre total de variables en mémoire si l'on déclarait 3 objets.
2. Identifier une fonction accesseur dans la classe Fleur et lui donner une implémentation.
3. Identifier et expliquer brièvement un exemple de polymorphisme dans la classe Fleur.
4. Dessiner un schéma simple pour illustrer l'état de la mémoire après l'exécution de chacune de ces instructions :

```
Fleur fleur1 = null, fleur2 = null;
fleur1 = new Fleur();
fleur2 = new Fleur();
fleur2 = new Fleur();
```

Exercice 07 :

Réaliser une classe *Point* permettant de représenter un point sur un axe. Chaque point sera caractérisé par un nom (de type *char*) et une abscisse (de type *double*). On prévoira :

•Un constructeur recevant en arguments le nom et l'abscisse d'un point,

•Une méthode *affiche* imprimant (en fenêtre console) le nom du point et son abscisse,

•Une méthode *translate* effectuant une translation définie par la valeur de son argument.

1. Écrire un petit programme utilisant cette classe pour créer un point, en afficher les caractéristiques, le déplacer et en afficher à nouveau les caractéristiques.
2. On voudrait aussi représenter ce point sur un plan bidimensionnel, sans modifier la classe précédente, rajouter le constructeur adéquat. Rajouter une méthode *translate* qui déplace l'abscisse de ce nouveau point de *dx*, son ordonnée de *dy*.

Exercice 08 :

Un feu de signalisation est défini par sa couleur, sa position et son hauteur. Le feu peut changer de couleur.

1. Définir la classe feu de signalisation, avec ses attributs et sa méthode change.
2. Une voiture est définie par son numéro d'immatriculation, sa marque, sa couleur et sa vitesse. Cette vitesse peut changer. Définir la classe Voiture.
3. Si le feu de signalisation devient vert, la voiture devant le feu (1iere) doit changer de vitesse à 50. Exprimer la nouvelle classe FeuDeSignalisation.
4. Écrire la méthode constructeur d'objets FeuDeSignalisation, permettant d'initialiser les attributs position et hauteur à des valeurs initiales et la couleur à verte.
5. Écrire l'instruction de création d'un objet NouveauFeu, de position 1 et de hauteur 4.