
PROGRAMMATION ORIENTÉE OBJET

TD1 : Classes et Objets(Pré-requis)

2022-2023

Exercice 1

Compléter les propositions suivantes relatives à la programmation orientée objet et au langage java :

- a) Pour empêcher l'accès direct à un attribut par un autre objet, on le fait précéder par le modificateur
- b) Le modificateur empêche la redéfinition de la méthode par les sous-classes.
- c) Pour permettre l'accès à une méthode par toutes les sous classes de la classe où elle est déclarée, on utilise le modificateur
- d) Un attribut dont la valeur est identique pour tous les objets de la classe doit être déclaré avec le modificateur
- e) Une méthode qui porte le nom de la classe est appelée
- f) int, double, boolean et char sont des types
- g) sert à déclarer un tableau v de réels à virgule flottante.
- h) sert à créer le tableau v composé de 20 réels à virgule flottante.
- i) La méthode..... est le point d'entrée d'un programme.
- j) Les nom de méthodes précédés par ne doivent retourner aucun résultat.

Exercice 2

Que fournit le programme suivant ?

```
1 class Sensor {
2     double temperature;
3 }
4 class TestSensor {
5     public static void main(String args[]) {
6         Sensor s1 = new Sensor();
7         Sensor s2 = new Sensor();
8         s1.temperature = 18.;
9         s2.temperature = 20.;
10        System.out.println((s1.temperature + s2.temperature)/2);
11    } }
```

Indiquer quel serait l'affichage produit si la variable *temperature* était définie en tant que variable de classe.

Exercice 3

Quelle erreur a été commise dans cette définition de classe ?

```
class ChCt
{ public ChCt (float r)
{ x = r ;
}
.....
private final float x ;
private final int n = 10 ;
private final int p ;
}
```

Exercice 4

Ce code compile t-il et, si non, quelle(s) erreur(s) le compilateur va t-il indiquer ?

```
class Toto{
int toto = 0;
Toto() {
toto = toto + 1;
}
public static void main(String[] tutu) {
Toto t1 = new Toto();
Toto t2 = new Toto();
System.out.println("Toto_:~" + toto);
}
}
```

Exercice 5

Le code suivant compile t-il ? Si non, indiquez les erreurs affichées par le compilateur et proposez des corrections. À quel affichage conduit l'exécution du programme (éventuellement corrigé) ?

```
class Test {
int i;
Test(int i) {
this.i = 12;
i = 15;
}
void i() {
Test i = new Test(3);
System.out.println(i.i);
i.i();
}
public static void main(String[] toto) {
Test i = new Test(34);
i.i();
}
}
```

Bonne chance !