

DEVOIR SURVEILLE

Matière : PROGRAMMATION ORIENTEE OBJET
Enseignantes : H. Mhafdh, ,S. Refai,, S.Majdoub
Classe : DSI2

Documents : Non autorisés
Durée : 1h
Date : Novembre 2023

*Le devoir comporte 3 pages
Il vous est demandé d'apporter un soin particulier à la présentation de votre copie.*

EXERCICE 1

(2 POINTS)

1) Soit les deux classes A et B définies comme suit

```
class A {  
    public int x;  
    public A() {x=5; }  
}  
  
class B extends A {  
    public B() {x++;}  
    public B(int i){this(); x=x+i; }  
    public B(String s){super(); x-=; }  
}
```

qu'affichera le code suivant?

```
B b1=new B(); B b2 =new B(2003); B b3= new B("Bonjour");  
System.out.println(b1.x + " et " + b2.x + " et encore " + b3.x );
```

- a) 6 et 2009 et encore 4
- b) 6 et 2004 et encore 4
- c) 6 et 2004 et encore 2003
- d) Autre

EXERCICE 2

(18 POINTS)

Une pépinière souhaite gérer sa collection de plantes en utilisant la programmation orientée objet.

Vous trouverez ci-dessous un exemple d'une méthode main suivi par son exécution :

```

public static void main(String[] args) {
    System.out.println("Pépinière GREEN HANDS");
    Pépinière pépinière = new Pépinière();
    Fleur rose = new Fleur(null, 30, 12, 15.99, "Rouge", 2);
    Fleur tulipe = new Fleur("Tulipe", 20, 8, 12.99, "Jaune", 4);
    Arbre chêne = new Arbre("Chêne", 200, 60, 89.99, "Caduque");
    Arbre sapin = new Arbre("Sapin", 150, 30, 59.99, "Persistant");
    rose.setNom("Rose");
    System.out.println("la "+rose.getNom()+" a le couleur "+rose.getCouleur());
    pépinière.ajoutPlante(rose);
    pépinière.ajoutPlante(tulipe);
    pépinière.ajoutPlante(chêne);
    pépinière.ajoutPlante(sapin);
    pépinière.afficherInventaire();
    System.out.println("Total Aborption CO2      "+ pépinière.totalAbsorptionTotaleCO2());
    System.out.println("Nombre d'arbres Caduques dans la pépinière : " + pépinière.compterArbresCaduques());
}

}

```

Ce code produira le résultat suivant :

```

Pépinière GREEN HANDS
la Rose a le couleur Rouge
Inventaire de la pépinière :
Plante #1
Nom : Rose: Hauteur : 30 cm : Âge : 12 Prix : 15.99 dinars Couleur : Rouge
La fleur n'est pas en fleurs.

Plante #2
Nom : Tulipe: Hauteur : 20 cm : Âge : 8 Prix : 12.99 dinars Couleur : Jaune
La fleur est en fleurs.

Plante #3
Nom : Chêne: Hauteur : 200 cm : Âge : 60 Prix : 89.99 dinars Type de feuillage : Caduque

Plante #4
Nom : Sapin: Hauteur : 150 cm : Âge : 30 Prix : 59.99 dinars Type de feuillage : Persistant

Total Aborption CO2      50.0
Nombre d'arbres Caduques dans la pépinière : 1

```

En s'inspirant de cet exemple, vous devez créer une hiérarchie de classes pour modéliser les différents types de plantes en ajoutant les méthodes nécessaires aux différentes classes afin d'avoir le résultat d'exécution ci-dessus.

Il vous a demandé de créer les classes suivantes

1- Une classe de base appelée **Plante** avec les attributs suivants

- nom (le nom de la plante)
- hauteur (la hauteur de la plante en centimètres)
- age (l'âge de la plante en mois)
- prix

a. Ajoutez une méthode **description** à la classe Plante qui affiche le nom, la hauteur, l'âge et le prix de la plante en faisant appel à la fonction **toString()** que vous devez implémenter.

3- Implémentez deux sous-classes de Plante :

- **Fleur** : Une Fleur est une Plante ayant en plus les champs **couleur (String)** et **moisF (int)** représentant le mois de florissant de la fleur.
 - a- Redéfinissez la méthode **description**
 - b- Ajoutez une méthode **void fleurir()** qui indique que la fleur est en fleurs ou non sachant que La fleur est considérée en fleurs si le moisF est 4, 5 ou 6
- **Arbre** : Un arbre est une Plante ayant en plus les attributs **type_Feuillage (String)** et **AbsorMoyenne** qui présente la moyenne d'absorption de CO2 par l'arbre : elle est fixée à 22 et ne peut pas être modifiée.
 - a- Redéfinissez la méthode **description**
 - b- Ajoutez une méthode **double absorptionCO2()** qui retourne la moyenne d'absorption de co2 ; si la taille de l'arbre dépasse 50 mètres, la moyenne d'absorption augmente de 3.

4- Créez une classe **Pépinière** pour gérer les plantes de la pépinière. Cette classe devrait inclure :

- Un attribut **inventaire** : tableau contenant les plantes que la pépinière propose à la vente.
 - Un attribut constant **MAX_PLANTES** qui vaut 1000 et représente le nombre maximum de plantes que la pépinière propose à la vente.
 - Un attribut **nombreDePlantes**: le nombre fictif de plantes ajoutés dans la pépinière. Initialisé à 0.
- a- Créez Un constructeur.
 - b- Ajoutez une méthode **ajoutPlante(Plante plante)** à la classe Pépinière qui ajoute une plante si possible. Sinon, elle affiche un message d'erreur
 - c- Ajoutez une méthode **afficherInventaire()** à la classe Pépinière qui affiche les détails de chaque plante, y compris son état de floraison ou la perte de ses feuilles le cas échéant.(voire l'exemple de la méthode main ci-dessus) pour les fleurs, affichez en plus leurs états de floraison.
 - d- Ajoutez une méthode **totalAbsorptionTotaleCO2()** qui retourne l'absorption de CO2 dans la pépinière.
 - e- Ajoutez une méthode **compterArbresCaduques()** qui retourne le nombre des arbres Caduque(type de feuillage caduque).