1 Présentation et écriture de complément pour une classe

Dans ce sujet, nous présentons quelques éléments d'un logiciel pour une agence événementielle, dont le métier est d'organiser des séminaires, des soirées, des salons, des animations pour des lancements de produits, etc. Dans ce sujet, nous nous concentrons sur quelques classes simplifiées pour représenter des organisations d'événements de type soirée avec repas et animation musicale ou ludique. Les noms des classes, attributs et méthodes vous sont indiqués en police de caractères `TrueType`. Une `AgenceEvenementielle` organise des `Evenements`. Un `Evenement` sera composé de plusieurs `Activites`. Nous considérons ici deux catégories d'activités : la `Restauration` et l'`AnimationMusicale`. Elles seront représentées comme des sous-classes de la classe `Activite`. Une classe incomplète pour représenter les agences événementielles est donnée ci-dessous.

```
public class AgenceEvenementielle {

private String nom = "nonRenseigne";

    // ... autres attributs (classe incomplète)

private static double tauxHonoraires = 9;
// 9% sur montant total de la valeur Hors Taxe (HT) d'une prestation

private static double TVAgenerale = 20; // 20% à l'heure actuelle
// la TVA vaut 20% sur toute activité économique
// sauf cas particuliers (alimentation, boisson)

public AgenceEvenementielle() {}
public AgenceEvenementielle(String nom) {this.setNom(nom);}

public String getNom() {return nom;}
public void setNom(String nom) {this.nom = nom;}

public static double getTauxHonoraires() {return AgenceEvenementielle.tauxHonoraires;}
public static void setTauxHonoraires(double nouveauTaux)
          {AgenceEvenementielle.tauxHonoraires = nouveauTaux; }

public static double getTVAgenerale() {return AgenceEvenementielle.TVAgenerale;}
public static void setTVAgenerale(double nouveauTaux)
           {AgenceEvenementielle.TVAgenerale = nouveauTaux;}
}
```

Question 1. Réécrivez la méthode `setTauxHonoraires` afin qu'elle n'accepte que des valeurs strictement positives pour le paramètre `nouveauTaux`. Lorsque la valeur du paramètre est négative ou nulle, elle affiche un message d'erreur et elle donne la valeur 1 à l'attribut.
Réponse - 1 pt

```
public static void setTauxHonoraires(double nouveauTaux){
   if (nouveauTaux > 0) // 0.5 pt (architecture de la conditionnelle)
         AgenceEvenementielle.tauxHonoraires = nouveauTaux;
   else
      {
         System.out.println("Erreur le taux doit être positif ou nul"); // 0.25 pt
         AgenceEvenementielle.tauxHonoraires = 1; // 0.25 pt
      }
}
```

Question 2. Si on décidait que le taux de la TVA générale ne peut jamais être modifié (elle serait constante), quelles modifications effectuerait-on sur la classe `AgenceEvenementielle` ?
Réponse - 1 pt

```
// QUESTION 2 (partie 1) on ajoute "final" // 0.5 pt
   private final static double TVAgenerale = 20;
// QUESTION 2 (partie 2) on enlève la méthode de modification // 0.5 pt
   // ici on enlève en la commentant
   // public static void setTVAgenerale(double nouveauTaux)
   //           {AgenceEvenementielle.TVAgenerale = nouveauTaux;}
```

2 Représentation des activités

On ne créera jamais d'instance directe de la classe `Activite`, seulement des instances de ses sous-classes.
Toutes les activités (`Activite`) sont décrites par :
— un nom (`nom`),
— le fait qu'elles se déroulent ou non en extérieur (`estExt`).
Une activité de restauration (`Restauration`) est une sorte d'activité avec les particularités suivantes :
— un nombre de convives (`nbConv`),
— un prix de repas par convive hors taxe (`prixR`),
— un prix pour la boisson par convive hors taxe (`prixB`),
Les prix pour les repas sont assujettis à une TVA particulière (`TVArepas`, 10% à l'heure actuelle) qui est la même pour toutes les activités de restauration (elle est partagée, mais non constante, elle peut en effet évoluer dans le temps). Les prix pour les boissons sont assujettis à la TVA générale (`TVAgenerale`) que vous avez vue apparaître dans la classe (`AgenceEvenementielle`). Vous devez choisir dans quelle classe mettre `TVArepas` et l'expliquer, plusieurs solutions sont acceptables.
Une animation musicale (`AnimationMusicale`) est une sorte d'activité avec les particularités suivantes :
— le prix du salaire des musiciens (`prixM`),
— le prix de la location du système de son (`prixS`).

Question 3. Ecrivez en Java les entêtes et attributs des classes représentant les activités. Pour la suite, vous supposerez qu'un constructeur sans paramètre et les accesseurs de type `get` et `set` existent pour tous les attributs de la classe (vous ne les écrivez pas).
Réponse - 3 pt

```
public abstract class Activite { //0.5 pt
   private String nom = "non renseigne"; //0.25 pt
   private boolean estExt; //0.5 pt
}

public class Restauration extends Activite{ //0.25 pt
   private int nbConv; //0.25 pt
   private double prixR, prixB; //0.25 pt

   private static double TVArepas = 10; /0.25 pt
   // cet attribut peut être mis ici ou dans AgenceEvenementielle
}

public class AnimationMusicale extends Activite { //0.25 pt
   private double prixM; //0.25 pt
   private double prixS; //0.25 pt
```

```
}
```

Question 4. Ecrivez en Java, pour les classes `Activite` et `Restauration` uniquement, un constructeur avec paramètres permettant d'initialiser les attributs propres aux instances (pas les éventuels attributs dont les valeurs sont partagées). Indiquez bien dans quelle classe vous écrivez quel constructeur.
Réponse - 2pt

```
// Dans Activite
public Activite(String n, boolean eE) { // 0.5pt
   this.setNom(n); this.setEstExt(eE); // 0.5pt
}
// Dans Restauration
public Restauration(String n, boolean eE, int nbc, double pR, double pB) { // 0.25pt
   super(n,ee); // 0.5pt
   this.setNbConv(nbc); // 0.25pt pour les trois instructions
   this.setPrixR(pR);
   this.setPrixB(pB);
}
```

Question 5. Ecrivez en Java, pour les classes `Activite` et `Restauration` uniquement, une méthode `void saisie(Scanner sc)` permettant la saisie de valeurs pour les attributs propres aux instances (pas pour les éventuels attributs dont les valeurs sont partagées). Indiquez bien dans quelle classe vous écrivez quelle méthode.
Réponse - 2 pt

```
// Dans Activite
public void saisie(Scanner sc) {
   System.out.println("Veuillez saisir le nom"); //0.25pt
   this.setNom(sc.next()); //0.25pt
   System.out.println("Veuillez saisir true ou false suivant si l'activite est en exterieur"); //0
   this.setEstExt(sc.nextBoolean()); //0.25pt
}
// Dans Restauration
public void saisie(Scanner sc) {
   super.saisie(sc); //0.5pt
   System.out.println("Veuillez saisir le nombre de convives"); //0.5pt pour tout le reste
   this.setNbConv(sc.nextInt());
   System.out.println("Veuillez saisir le prix du repas / convive");
   this.setPrixR(sc.nextDouble());
   System.out.println("Veuillez saisir le prix de la boisson / convive ");
   this.setPrixB(sc.nextDouble());
}
```

Question 6. Ecrivez en Java, pour les classes `Activite` et `Restauration` uniquement, une méthode `prixTTC`, retournant le prix TTC (Toutes Taxes Comprises). Indiquez bien dans quelle classe vous écrivez quelle méthode.

- Le prix TTC d'une activité en général n'est pas défini.
- Le prix TTC d'une activité de restauration est la somme (1) du prix TTC des repas (produit du nombre de convives par le prix hors taxe d'un repas par convive, sur lequel on applique la TVA appropriée `TVArepas`) et (2) du prix TTC des boissons (produit du nombre de convives par le prix hors taxe par convive des boissons , sur lequel on applique la TVA appropriée (`TVAgenerale`)).

Réponse - 2 pt

```
// Dans Activite
   public abstract double prixTTC(); // 0.25 pt signature + 0.5 pt abstract
// Dans Restauration
   public double prixTTC() {
      return  (this.nbConv * this.prixR) * (1+Restauration.TVArepas/100) // 0.5 pt
            + (this.nbConv * this.prixB)
                        * (1+AgenceEvenementielle.getTVAgenerale()/100); // 0.75 pt
   }
```

3 Représentation des événements

Un événement (`Evenement`) a un nom et se compose de plusieurs activités.

Question 7. Ecrivez en Java l'entête et les attributs de la classe `Evenement` en les initialisant.
Réponse - 1.5 pt

```
public class Evenement { / 0.25
   private String nom = "non renseigne"; // 0.25
   private ArrayList<Activite> listeActivites = new ArrayList<Activite>(); // 1 pt
}
```

Question 8. Ecrivez en Java, dans la classe `Evenement`, une méthode `ajout` permettant de lui ajouter une activité, à la condition que celle-ci n'y soit pas déjà prévue (sinon le message d'erreur "activité déjà présente" est affiché). Si l'activité est ajoutée, on affiche le message "activité ajoutée".
Réponse - 1.5 pt

```
public void ajout(Activite a) { // 0.25 pt
   if (this.listeActivites.contains(a)) // 0.25 pt
         System.out.println("activité déjà présente"); // 0.25 pt
   else // 0.25 pt
      {
         this.listeActivites.add(a); // 0.25 pt
         System.out.println("activité ajoutée"); // 0.25 pt
      }
}
```

Question 9. Ecrivez en Java, dans la classe `Evenement`, une méthode `prixTTC` retournant le prix toutes taxes comprises de l'événement. Ce prix se calcule comme la somme des prix TTC des activités composant l'événement à laquelle on ajoute les honoraires de l'agence événementielle. Ces honoraires correspondent à un certain pourcentage de la somme du prix TTC des activités, précisé dans l'attribut `tauxHonoraires`.
Réponse - 3 pt

```
public double prixTTC() { // 0.5 pt
   double res = 0; // 0.5 pt
   for (Activite a : this.listeActivites) // 0.5 pt
      res+=a.prixTTC(); // 0.5 pt
   return res*(1+AgenceEvenementielle.getTauxHonoraires()/100); // 1 pt
}
```

Question 10. Ecrivez en Java, dans la classe `Evenement`, une méthode `activExt` retournant une liste de chaînes de caractères composée des **noms des activités** se déroulant en extérieur.
Réponse - 3 pt

```
public ArrayList<String> activExt() { // 0.75 pt
   ArrayList<String> res = new ArrayList<String>(); // 0.5 pt
   for (Activite a : this.listeActivites) // 0.5 pt
         if (a.isEstExt()) // 0.5 pt
            res.add(a.getNom()); // 0.5 pt
   return res; // 0.25 pt
}
```