

TP1 BIS

- PROGRAMMATION ORIENTÉE OBJET EN JAVA -

Exercice 1 (conversion explicite)

Écrire un programme calculant la somme des n premiers termes de la "série harmonique", c'est-à-dire la somme :

$$1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$$

La valeur de n sera lue en donnée^a.

Exercice 2 (switch)

Écrire un programme java qui selon le mois (saisi au clavier) détermine la saison correspondante. Utiliser la structure conditionnelle multiple "selon" comme suit:

- une instruction qui affiche la saison
- une expression retourne la saison

il faut prendre en compte la casse de la chaîne saisie par l'utilisateur.

Exercice 3 (boucles)

Écrire un programme qui affiche un triangle isocèle formé d'étoiles. La hauteur du triangle (c'est-à-dire son nombre de lignes) sera fourni en donnée^a, comme dans l'exemple ci-dessous. On s'arrangera pour que la dernière ligne du triangle s'affiche sur le bord gauche de l'écran.

```
combien de lignes ? 5
  *
 ***
*****
*****
*****
```

Exercice 4 (classe simple)

Un élève sera ici modélisé par la classe Eleve d'un paquetage nommé gestionEleves, de la façon suivante.

La classe Eleve possède trois attributs privés :

- son nom, nommé nom, de type String,
- un ensemble de notes, nommé listeNotes, qui sont des réels rangés dans un tableau (Chaque élève possède au maximum 10 notes, en cas d'absence de note un 0 est attribué)
- une moyenne de type double, nommée moyenne, qui doit toujours être égale à la moyenne des notes contenues dans l'attribut listeNotes. Un élève sans aucune note sera considéré comme ayant une moyenne nulle.

La classe Eleve possède un constructeur permettant uniquement d'initialiser le nom de l'élève.

La classe Eleve possède aussi cinq méthodes publiques :

- Un getter pour la moyenne de l'élève: ***public double getMoyenne()*** renvoie la valeur de l'attribut moyenne ;
- Un getter pour le nom de l'élève : ***public String getNom()*** renvoie le nom de l'élève ;
- Un getter pour la liste des notes de l'élève: ***public double[] getListeNotes()*** renvoie la liste des notes de l'élève ;
- La méthode d'en-tête: ***public void ajouterNote(double note)*** ajoute la note reçue en paramètre à listeNotes ; si la note reçue en paramètre est négative, la note introduite est 0 ; si la note reçue en paramètre est supérieure à 20, la note introduite est 20 ; la méthode actualise en conséquence l'attribut moyenne.
- La méthode d'en-tête: ***public String toString()*** qui retourne une description de l'élève considéré (par exemple : "Ahmed (12.25)").

Après avoir terminé la classe Eleve, écrire un programme qui teste cette classe.