TP1: Notions de base	-AU.2022/2023-	Module : Prog_Java

Exercice 1.

Le but de cet exercice est de se familiariser avec la méthode `main` pour le lancement de programmes, les sorties standard `System.out.println`.

1. Écrivez une classe, appelée `PremiersPas`, qui ne contient qu'une méthode, la méthode main, dans laquelle vous afficherez une chaîne de caractères quelconque (par exemple le classique *« Salamo Alikoum »*). Tapez le code dans un fichier au nom approprié, compilez et exécutez.
2. Ajouter une instruction qui afficher les nombres paire existant entre 0 et 20

Exercice 2.

Le but de cet exercice est de se familiariser avec la classe Scanner, qui vous est fournie pour permettre des entrées de données simplifiées.

1. En utilisant la classe Scanner écrivez un programme pour afficher la question *« Fait-il beau ? »*, attendre une réponse « oui » ou *« non »*. Si la réponse est *« oui »*, le programme affichera *« Tant mieux ! »*, si c'est *« non »*, il affichera *« Espérons que ce soit mieux demain »*, et si ce n'est ni l'un ni l'autre, *« je ne comprends pas ! »*.
2. Modifiez le programme précédent pour que, tant que la saisie de l'utilisateur est incorrect, le programme affiche *« merci de répondre par oui ou par non »* et redemande une réponse (accepter aussi *« o »* et *« n »* comme réponses valides

Exercice 3.

Le but de cet exercice est de vous initier à la programmation POO

1. Écrivez une classe **Point** qui représente un point dans le plan en coordonnées cartésiennes. Cette classe devra avoir deux attributs de type double nommés x et y.
2. Ajoutez à la classe Point un constructeur afin de permettre l'initialisation de x et y.
3. Ajouter à la classe Point une méthode double **distanceTo**(Point p) qui retourne la distance du point au point p. Utiliser double Math.sqrt(double d) pour calculer la racine carrée.
4. Ajouter à la classe Point une méthode Point **translate**(double dx, double dy) qui retourne un point obtenu par translation du point de dx sur les abscisses et de dy sur les ordonnées.
5. Écrivez une classe **LineSegment** qui possède deux propriétés point1 et point2 de type Point représentant les deux extrémités d'un segment.
6. Ajoutez à la classe LineSegment un constructeur afin de permettre l'initialisation de point1 et point2.
7. Ajouter à la classe LineSegment une méthode double length() qui calcule et retourne la longueur du segment. Est-ce qu'une partie du calcul peut être déléguée à une méthode de la classe point ?