TP2

Object-oriented programming

(Classes, objetcs, constructors)

Exercise 01 : (Simple class)

Create a class Point to represent a point on an axis. Each point will be characterized by a name (of type char) and an abscissa (of type double).

Provide:

• A constructor that takes the name and abscissa of a point as arguments,

• A method 'display' that prints (in the console window) the name of the point and its abscissa,

• A method 'translate' that performs a translation defined by the value of its argument.

Write a small program using this class to create a point, display its characteristics, move it, and then display its characteristics again

Exercice 02 : (Simple class distance)

Create a class called 'Distance'. This class should have 2 integers (x, y) members. The constructor should set the values of x and y through parameters. The class should have a public method called 'distanceCalcule'. It takes a single parameter (Distance) and returns the distance between the two points.

Example:

P1 (5,6)

P2 (3,2)

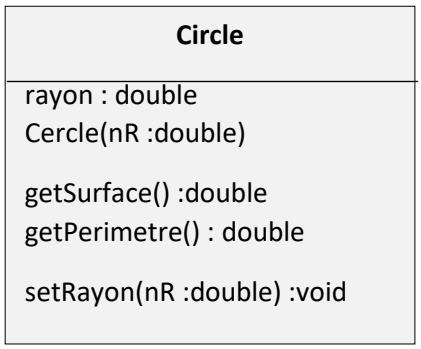
The distance between P1 and P2 is : 4.47214

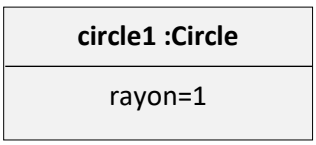
Note :

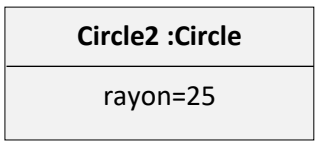
$$Distance = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

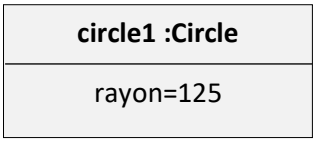
Exercise 03: (Circle class)

- Provide the Java code for the Circle class of the following UML diagram:









- Write the program Td2Circle that uses the Circle class to create 3 objects with respective radius of 1, 25, and 125, and allows displaying the radius, perimeter and surface of each circle. Then, the program changes the radius of the second object to 100 and displays its radius, perimeter and surface.

Exercise 04 : (Matrix class)

Create a Java program that defines a class named **Matrice** with the following specifications:

1. Attributes:
    - **tab**: a 2-dimensional integer array with dimensions 2x3.
2. Methods:
    - **Read_mat**: a method that reads values into the **tab** array from the user.
    - **Write_mat**: a method that prints the elements of the **tab** array.
    - **Sum_mat**: a method that calculates and returns the sum of all elements in the **tab** array.
    - **Max_mat**: a method that finds and returns the maximum value in the **tab** array.

Exercise 05 : (Tv class)

Provide the Java code for the TV class, whose UML representation is as follows, and propose a program that will use it.

TV
Channel :int
volume : int
on : boolean
TV()
turnOn() :void
turnOff() : void
setChanel(nCh :int) :void
setVolume(nVol) :void
chanSup() :void
chanInf() :void
volumeSup() :void
volumeInf() :void

The channels Tv(1 TO 120)

The volume (1 to 7)

Increment The no of the channel by 1

Decrement The no of the channel by 1

Increment the volume by 1

Decrement the volume by 1

Excercise 06 : (Multiplication quiz)

The multiplication quiz program, the Java code of which follows, generates the execution of figure bellow. Provide the Java code for the Quiz class and its method calculScore

```
import java.util.Scanner;
public class MultiQuiz {
   public static void main(String[] args) {
      String name;
      int num = 0;
// Prompt the candidate to enter their name and a number
      Scanner sc = new Scanner(System.in);
      System.out.println("\nEnter your name: ");
      name = sc.next();
      System.out.println("\nEnter the quiz number: ");
      num = sc.nextInt();
      Quiz Q = new Quiz(name, num);
      Q.calculScore();
   }
}
```

```
Enter your name : Djouama

Enter the nombre of the quiz :7

1 X 7 = 7

Correct !

2 X 7 = 27

False !

….

Djouama ! Your score is 8/10
```