**Lab 5. Coding assignment**

**5.1 Create a class Point2D , : for representing a point in x-y co-ordinate system.**

**5.2 Create a parameterized constructor to init x & y co-ords.**

**5.3 Add a method to return string form of point's x & y co-ords**

**Hint : public String toString())**

**5.4 Add isEqual method to Point2D class :a boolean returning method : must return true if both points are having same x,y co-ords or false otherwise.**

**eg : public boolean isEqual(Point2D anotherPoint)**

**{**

**.......**

**}**

**eg : p1.isEqual(p2)**

**5.5 Add calculateDistance method to calculate distance between current point and specified point & return the distance to the caller.**

**Hint : Use distance formula . Use java.lang.Math class methods --sqrt, pow etc.**

**eg : public double calculateDistance(Point2D anotherPoint)**

**{**

**Math.sqrt(.....);**

**}**

**5.6 Write TestPoint class with a main method**

**Accept co ordinates of 2 points from user (Scanner) --to create 2 points (p1 & p2)**

**5.7 Use getDetails method to display point details.(p1's details & p2's details)**

**5.8 Invoke isEqual & display if points are same or different (i.e p1 & p2 are located at the same position)**

**5.9 Display distance between p1 & p2**

**Program :**

**Class Point2D**

package com.code;

import java.lang.Math;

public class Point2D {

private double x;

private double y;

public Point2D() {

}

public Point2D(double x, double y) {

this.x = x;

this.y = y;

}

@Override

public String toString() {

return "x : " + x + " y : " + y;

}

public boolean isEqual(Point2D obj) {

if (this.x == obj.x && this.y == obj.y)

return true;

else

return false;

}

public double calculateDistance(Point2D obj1) {

double dist = Math.*sqrt*((this.x - obj1.x) \* (this.x - obj1.x) + (this.y - obj1.y) \* (this.y - obj1.y));

return dist;

}

}

**Class Point2DTest**

package com.tester;

import java.util.Scanner;

import com.code.Point2D;

public class Point2DTest {

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

Point2D p1 = new Point2D();

System.*out*.println("Enter points");

Point2D p2;

p2 = new Point2D(sc.nextDouble(), sc.nextDouble());

Point2D p3;

p3 = new Point2D(sc.nextDouble(), sc.nextDouble());

System.*out*.println(p1.toString());

System.*out*.println(p2.toString());

System.*out*.println(p3.toString());

boolean result = p2.isEqual(p3);

System.*out*.println(result);

double value = p2.calculateDistance(p3);

System.*out*.println(value);

}

}

**Output :**

