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

Adv Lab.

Create Array Of Point2D and store 5 objects in it;

// **package** com.util;

**package** com.util;

**import** java.lang.Math;

**public** **class** Point2D {

**private** **float** x;

**private** **float** y;

**public** Point2D(**float** x, **float** y) {

**this**.x=x;

**this**.y=y;

}

**public** **void** getDetails() {

System.***out***.println("Point2D = X :"+x+" Y :"+y);

}

**public** **boolean** isEquals(Point2D obj) { // obj1.isEquals(obj2)

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

**return** **true**;

}

**return** **false**;

}

**public** **double** calculateDistance(Point2D obj) {

**return** Math.*sqrt*(Math.*pow*((**this**.x-obj.x),2)+Math.*pow*((**this**.y-obj.y),2));

}

}

//**package** com.tester;

**package** com.tester;

**import** java.util.Scanner;

**import** com.util.Point2D;

**public** **class** TesterPoint {

**public** **static** **void** main(String[] args) {

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

Point2D[] point=**new** Point2D[5];

**int** ch,index=0;

**do** {

System.***out***.println("Select 1.add point 2.display Point 3.Find Distance 4.compare 5.exit");

System.***out***.print("Enter choice :");

ch=sc.nextInt();

**switch**(ch) {

**case** 1:

System.***out***.println("Enter X and Y co-ordinates : ");

Point2D p=**new** Point2D(sc.nextFloat(),sc.nextFloat());

point[index]=p;

index++;

**break**;

**case** 2:

**for**(**int** i=0;i<5;i++) {

**if**(point[i]!=**null**)

point[i].getDetails();

**else**

**break**;

}

**break**;

**case** 4:

System.***out***.println("Enter co-ordinates X and Y : ");

Point2D p1=**new** Point2D(sc.nextFloat(),sc.nextFloat());

System.***out***.println("Enter co-ordinates X and Y : ");

Point2D p2=**new** Point2D(sc.nextFloat(),sc.nextFloat());

**if**(**true** == p1.isEquals(p2)) {

System.***out***.println("Points are Equal");

} **else** {

System.***out***.println("Points are Not Equal");

}

**break**;

**case** 3:

System.***out***.println("Enter co-ordinates X and Y : ");

Point2D pa=**new** Point2D(sc.nextFloat(),sc.nextFloat());

System.***out***.println("Enter co-ordinates X and Y : ");

Point2D pb=**new** Point2D(sc.nextFloat(),sc.nextFloat());

System.***out***.println(pa.calculateDistance(pb));

**break**;

**case** 5:

System.***out***.println("Exit");

**break**;

**default**:

System.***out***.println("Invalid Choice");

**break**;

}

} **while**(ch!=5);

sc.close();

}

}



