

ArrayList

1. The following program utilizes the ArrayList and Point2D.Double classes. The Point2D.Double class encapsulates X and Y coordinate values into a single point object (storing each value as a double). ArrayList is then used to create a list of coordinate points. For more information on the usage of Point2D.Double, as well as other Java classes, refer to:

<https://docs.oracle.com/javase/7/docs/api/java/awt/geom/Point2D.Double.html>

Explore this code and determine its exact output if the following points are read from the keyboard:
 $(1.5, 2.1), (9.7, 2.1), (9.7, 7.3), (1.5, 7.3)$.

These points represent consecutive vertices of a rectangle and must be entered in consecutive order!

```

import chn.util.*;      // for ConsoleIO
import apcslib.*;      // for formatting
import java.awt.geom.*; // for Point2D.Double
import java.util.*;      // for ArrayList

class Rectangle
{
    public Rectangle()
    {
        ArrayList myRectangle = new ArrayList();
        input(myRectangle);
        output(myRectangle);
        double area = calculateArea(myRectangle);
        System.out.println("The area of the rectangle is "
            + Format.right(area,5,2));
    }

    public void input(ArrayList myRect)
    {
        System.out.println("We need four vertices for our rectangle.");
        System.out.println("Please provide them in consecutive order.");
        ConsoleIO console = new ConsoleIO();
        for (char ch = 'A'; ch <= 'D'; ch++)
        {
            System.out.print("Give me the x coordinate for point " + ch + ": ");
            double x = console.readDouble();
            System.out.print("Give me the y coordinate for point " + ch + ": ");
            double y = console.readDouble();
            Point2D.Double myPoint = new Point2D.Double(x,y);
            myRect.add(myPoint);
        }
    }
}

```

```

public void output(ArrayList myRect)
{
    for (char ch = 'A'; ch <= 'D'; ch++)
    {
        Point2D.Double pt = (Point2D.Double)myRect.get(ch - 65);
        System.out.println("Point " + ch + " is (" + pt.getX() + ", "
            + pt.getY() + ")");
    }
}

public double calculateArea(ArrayList myRect)
{
    Point2D.Double ptA = (Point2D.Double)myRect.get(0);
    Point2D.Double ptB = (Point2D.Double)myRect.get(1);
    Point2D.Double ptC = (Point2D.Double)myRect.get(2); // not needed
    Point2D.Double ptD = (Point2D.Double)myRect.get(3);
    double base = ptA.distance(ptB);
    double height = ptA.distance(ptD);
    return base * height;
}

public static void main(String[] args)
{
    Rectangle app = new Rectangle();
}
}

```

2. Revise this program by adding the methods *input5thPoint* and *calculateTriangleArea*. The method *input5thPoint* should prompt the user to add a 5th point on the line segment connecting points A and B (i.e. the first two points entered). Within this method, the point should be added to the ArrayList *myRectangle*. The method *calculateTriangleArea* should calculate the area of the triangle formed by this 5th point and the points A and D.