

1 Instructions

As usual you will work in pairs and only one of you needs to submit the files. Here we give you the exact main program that you must use and your task is to create the methods needed to complete the program. Submit the Java code files containing the program and class. Also submit a Word, LibreOffice, or text file containing at least 3 runs of the program with different heights and widths. These runs should test all possible situations that can arise, so if 3 is not enough to do this you will need to do more.

2 Exercise

This exercise is to create another class structure (Object). Create a new project and a new main class as usual. Now create another class called Rectangle.

In the Rectangle class:

1. Have two private data members `width` and `height`, both doubles.
2. Create a constructor that brings in as parameters doubles for the height and width and sets the two data members to the parameter values. You do not need to worry about data validity checking.
3. Create sets and gets accessor methods that will set and return the values of the height and width. Four methods total here. Again, you do not need to worry about data validity checking.
4. Create a method `Perimeter` that will calculate and return the value of the perimeter of the rectangle.
5. Create a method `Area` that will calculate and return the value of the area of the rectangle.
6. Create a method `isSquare` that will return true if the rectangle is a square and false if it is not.
7. Create a method `toString` that will return a string with the height and width information. Specifically, if we have a rectangle object called `rectangle1` with width 7 and height 2 the code line

```
System.out.println(rectangle1);
```

would produce the output of

```
Rectangle Data: Width = 7.0 Height = 2.0
```

In the main file:

1. Create the following method. Note that this method brings in a Rectangle object as a parameter. As we discussed in class, these objects are new data types you create and can be used like any other data type. So sending a Triangle or Rectangle to a method is just as easy as sending an int or a double.

```
public static void PrintRectangleInformation(Rectangle r) {  
    System.out.println("Height: " + r.getHeight());  
    System.out.println("Width: " + r.getWidth());  
    System.out.println("Area: " + r.Area());  
    System.out.println("Perimeter: " + r.Perimeter());  
}
```

```

    if (r.isSquare())
        System.out.println("The rectangle is a square.");
    else
        System.out.println("The rectangle is not a square.");
}

```

2. Insert the following code into the main itself,

```

Scanner kb = new Scanner(System.in);

System.out.print("Input height and width of Rectangle #1: ");
double h = kb.nextDouble();
double w = kb.nextDouble();

Rectangle rectangle1 = new Rectangle(h, w);

System.out.print("Input height and width of Rectangle #2: ");
h = kb.nextDouble();
w = kb.nextDouble();

Rectangle rectangle2 = new Rectangle(h, w);

System.out.println();
System.out.println(rectangle1);
System.out.println(rectangle2);

System.out.println();
System.out.println("Rectangle 1");
PrintRectangleInformation(rectangle1);

System.out.println();
System.out.println("Rectangle 2");
PrintRectangleInformation(rectangle2);

rectangle1.setHeight(17);
rectangle1.setWidth(17);

System.out.println();
System.out.println("Rectangle 1");
PrintRectangleInformation(rectangle1);

```

Run the program and you should get output like the following. You may need to debug your Rectangle class if you have any errors.

```

Input height and width of Rectangle #1: 5 7
Input height and width of Rectangle #2: 6 6

Rectangle Data: Width = 7.0    Height = 5.0
Rectangle Data: Width = 6.0    Height = 6.0

Rectangle 1
Height: 5.0
Width: 7.0
Area: 35.0
Perimeter: 24.0
The rectangle is not a square.

Rectangle 2
Height: 6.0
Width: 6.0
Area: 36.0
Perimeter: 24.0
The rectangle is a square.

Rectangle 1
Height: 17.0
Width: 17.0
Area: 289.0
Perimeter: 68.0
The rectangle is a square.

```
