Circle.java | Comparable.java | Resizable.java | ResizeAndCompareShape.jav

```java
package lab10;

public abstract class Shape {
    protected String color;
    abstract double getArea();
    public void setColor(String color) {
        this.color = color;
    }
    public String toString() {
        return "color : "+ color;
    }
    public int compareTo(Object object) {
        if (this.getArea() > ((Square)object).getArea())
            return 1;
        else if (this.getArea() < ((Square)object).getArea())
            return -1;
        else
            return 0;
    }
}

```

Circle.java | Comparable.java | Resizable.java | Resize

```
1  package lab10;
2
3  public interface Comparable {
4      public int  ComparableTo(Object object);
5  }
6
```

Circle.java | Comparable.java | Resizable.java

```
1  package lab10;
2
3  public interface Resizable {
4      public void resize(int percent);
5
6  }
7
```

```java
package lab10;

public class Circle extends Shape{
    protected double radius;
    public Circle(String color,int radius) {
        super.setColor(color);
        this.radius = radius;
    }
    public double getArea() {
        return 3.14*radius*radius;

    }
    public double getCirle() {
        return 2*3.14*radius;
    }
    public String toString() {
        return super.toString()+"radius : "+radius+"\nArea ; "+ (int)getArea()+",Circumference : "+(int)getCirle();
    }
    public void resize(int percent) {
        this.radius = this.radius*(percent/100);
    }
}

```

```
    public Square(String color,int width) {
        super.setColor(color);
        this.width = width;
    }
    public double gatArea() {
        return width*width;
    }
    public double gatPerimeter() {
        return 4*width;
    }
    public String toString() {
        return super.toString()+"width : "+width+"\nArea ; "+ (int)getArea()+",Perimeter : "+(int)gatPerimeter();
    }
    public void resize(int percent) {
        this.width = this.width*(percent/100);
    }
}
```

```java
package lab10;

import java.util.ArrayList;

public class ResizeAndCompareShape {
    public static void mian(String[] args) {
        ArrayList<Shape>mix = new ArrayList<Shape>();
        mix.add(new Square("Orange",4));
        mix.add(new Square("Orange",10));
        mix.add(new Square("Orange",5));
        mix.add(new Circle("Violent",3));
        mix.add(new Circle("Violent",6));
        mix.add(new Circle("Violent",10));
        displayAllShape(mix);
        for(int i=0;i<mix.size();i++) {
            if (mix.get(i)instanceof Square)
                ((Square)mix.get(i)).resize(200);
            else
                ((Circle)mix.get(i)).resize(50);
        }
        System.out.println("After resize");
        displayAllShape(mix);
        int comp1 = mix.get(0).compareTo(mix.get(1));
        String[] compMessage = {"smaller","epual","bigger"};
        System.out.println("Compare 2 Squares after resize");
        System.out.println("Square 1 is"+compMessage[comp1+1]+" than Square 2");
        int comp2 = mix.get(1).compareTo(mix.get(0));
        System.out.println("Compare 2 Circle after resize");
        System.out.println("Square 1 is"+compMessage[comp2+1]+" than Square 2");
        }
    public static void displayAllShape(ArrayList<Shape> A) {
        System.out.println("Class Square");
        for (int i = 0;i< A.size();i++) {
            if(A.get(i)instanceof Square)
                System.out.printf("%d. %s\n",(i+1),((Square)A.get(i)).toString());
        }
        System.out.println("============================================================");
        System.out.println("Class Circle");
        int i, j=0;
```

```
40          for (i = 0;i<A.size();i++) {
41              if(A.get(i)instanceof Circle)
42                  System.out.printf("%d.%s\n",(i+1),((Circle)A.get(i)).toString());
43                  j++;
44          }
45
46  }
47  }
```

```
Class Square
1. color : Orange width : 4
Area : 16, Perimeter : 16
2. color : Orange width : 10
Area : 100, Perimeter : 40
3. color : Orange width : 5
Area : 25, Perimeter : 20
================================================
Class Circle
4. color : Violent radius : 3.0
Area : 28, Circumference : 18
5. color : Violent radius : 6.0
Area : 113, Circumference : 37
6. color : Violent radius : 10.0
Area : 314, Circumference : 62
================================================
After resize
Class Square
1. color : Orange width : 8
Area : 64, Perimeter : 32
2. color : Orange width : 20
Area : 400, Perimeter : 80
3. color : Orange width : 10
Area : 100, Perimeter : 40
================================================
Class Circle
4. color : Violent radius : 1.5
Area : 7, Circumference : 9
5. color : Violent radius : 3.0
Area : 28, Circumference : 18
6. color : Violent radius : 5.0
Area : 78, Circumference : 31
================================================
Compare 2 Squares after resize
Square 1 is smaller than Square 2
Compare 2 Circle  after resize
Circle  1 is bigger than Circle 2
```

Point2D.java MovableLine.java Movable.java

```java
package lab10;

public interface Movable {
    public void moveUp(int y);
    public void moveDown(int y);
    public void moveLeft(int x);
    public void moveRight(int x);
}

```

Point2D.java | MovableLine.java | Movable.java

```java
package lab10;
public class Point2D implements Movable {
    private int x;
    private int y;
    public Point2D() {
    }
    public Point2D (int x, int y) {
        this.x = x;
        this.y = y;
    }

    public int getX() {
        return this.x;
    }

    public int getY() {
        return this.y;
    }

    public void setX(int x) {
        this.x = x;
    }

    public void setY(int y) {
        this.y = y;
    }

    public String toString() {
        return "From ("+this.x+ ","+this.y+") " ;
    }

    public void moveUp(int y) {
        this.y = this.y + y;
    }

    public void moveDown(int y) {
        this.y = this.y - y;
    }

```

Point2D.java | MovableLine.java | Movable.java

```java
package lab10;
import java.util.Scanner;

public class MovableLine {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Please enter number of line: ");
        int n = sc.nextInt();
        int size = n*2;
        System.out.println("You need "+size+" Points to formed "+n+" Lines. ");
        Point2D[] line = new Point2D[size];
        line[0] = new Point2D(5,7);
        line[1] = new Point2D(10,12);
        line[2] = new Point2D(20,12);
        line[3] = new Point2D(15,22);
        line[4] = new Point2D(30,10);
        line[5] = new Point2D(24,20);
        displayAll(line);

        System.out.println("============================");
        line[0].moveLeft(10);
        line[0].moveUp(20);
        line[1].moveLeft(10);
        line[1].moveUp(20);
        line[2].moveRight(30);
        line[2].moveDown(100);
        line[3].moveRight(30);
        line[3].moveDown(100);
        displayAll(line);
    }

    public static void displayAll(Point2D[] line) {
        double x1 =0,x2 =0,y1 =0, y2 =0, s =0, d =0;
        int i =0, j =1;
        for (i =0;i < line.length;i+=2) {
            x1 = line[i].getX();
            y1 = line[i].getY();
            x2 = line[i+1].getX();
            y2 = line[i+1].getY();
```

```
32      public static void displayAll(Point2D[] line) {
33          double x1 =0,x2 =0,y1 =0, y2 =0, s =0, d =0;
34          int i =0, j =1;
35          for (i =0;i < line.length;i+=2) {
36              x1 = line[i].getX();
37              y1 = line[i].getY();
38              x2 = line[i+1].getX();
39              y2 = line[i+1].getY();
40              s = (y1 - y2)/ (x1 - x2);
41              d = Math.sqrt((y2 - y1) * (y2 - y1) + (x2 -x1) * (x2 - x1));
42              System.out.println("Line no. "+j+ " From "+line[i].toString() + " to " + line[i+1].toString());
43              System.out.printf("Slope is %.6f Distant is %.6f \n\n",s,d);
44              j++;
45          }
46      }
47  }
```

```
<terminated> MovableLine [Java Application] C:\Users\Lenovo\Desktop\eclipse\plugi
Please enter number of line: 3
You need 6 Points to formed 3 Lines.
Line no. 1 From From (5,7)  to From (10,12)
Slope is 1.000000 Distant is 7.071068

Line no. 2 From From (20,12)  to From (15,22)
Slope is -2.000000 Distant is 11.180340

Line no. 3 From From (30,10)  to From (24,20)
Slope is -1.666667 Distant is 11.661904

==============================
Line no. 1 From From (-5,27)  to From (0,32)
Slope is 1.000000 Distant is 7.071068

Line no. 2 From From (50,-88)  to From (45,-78)
Slope is -2.000000 Distant is 11.180340

Line no. 3 From From (30,10)  to From (24,20)
Slope is -1.666667 Distant is 11.661904
```