Lab2GeometricShape(Test)

```
package lab2geometricshape;


public class Lab2GeometricShape {

    public static void main(String[] args) {


        Circle circle1 = new Circle();
        Circle circle2 = new Circle(3.0);

        System.out.println("Area of Circle circle2 is " + circle2.getArea());
        System.out.println("Perimeter of Circle circle2 is " + circle2.getPerimeter());


        if (circle1.compareTo(circle2) == 0)

            System.out.println("Circle circle1 and circle2 have equal coverage of area");

         else if (circle1.compareTo(circle2) > 0)

            System.out.println("Circle circle1 has larger area than the circle circle2");

         else

            System.out.println("Circle circle1 has smaller area than the circle circle2");

        Circle circle3 = circle2.clone();
        if (circle2.compareTo(circle3) == 0)

            System.out.println("Circle circle2 and circle3 have equal coverage of area");

         else if (circle2.compareTo(circle3) > 0)

            System.out.println("Circle circle2 has larger area than the circle circle3");

         else

            System.out.println("Circle circle2 has smaller area than the circle circle3");

        System.out.println("Circle circle3: " + circle3);
```

```
System.out.println("-----------------------------------------------------------------");

Ellipse ellipse1 = new Ellipse(2.0,2.0);
Ellipse ellipse2 = new Ellipse(3.0, 3.0);

  System.out.println("Area of Ellipse ellipse2 is " + ellipse2.getArea());
  System.out.println("Perimeter of Ellipse ellipse2 is " + ellipse2.getPerimeter());

 if (ellipse1.compareTo(ellipse2) == 0)

    System.out.println("Ellipse ellipse1 and ellipse2 have equal coverage of area");

 else if (ellipse1.compareTo(ellipse2) > 0)

    System.out.println("Ellipse ellipse1 has larger area than the circle ellipse2");

 else

    System.out.println("Ellipse ellipse1 has smaller area than the ellipse ellipse2");

Ellipse ellipse3 = ellipse2.clone();

if (ellipse2.compareTo(ellipse3) == 0)

    System.out.println("Ellipse ellipse2 and ellipse3 have equal coverage of area");

 else if (ellipse2.compareTo(ellipse3) > 0)

    System.out.println("Ellipse ellipse2 has larger area than the ellipse ellipse3");

 else

    System.out.println("Ellipse ellipse2 has smaller area than the ellipse ellipse3");

System.out.println("Ellipse ellipse3: " + ellipse3);

System.out.println("-----------------------------------------------------------------");


Octagon octagon1 = new Octagon();
Octagon octagon2 = new Octagon(3.0);
```

```
System.out.println("Area of Octagon octagon2 is " + octagon2.getArea());
System.out.println("Perimeter of Octagon octagon2 is " + octagon2.getPerimeter());

if (octagon1.compareTo(octagon2) == 0)

   System.out.println("Octagon ooctagon1 and octagon2 have equal coverage of area");

 else if (octagon1.compareTo(octagon2) > 0)

   System.out.println("Octagon octagon1 has larger area than the Octagon octagon2");

 else

   System.out.println("Octagon octagon1 has smaller area than the Octagon octagon2");

Octagon octagon3 = octagon2.clone();
if (octagon2.compareTo(octagon3) == 0)

   System.out.println("Octagon octagon2 and octagon3 have equal coverage of area");
 else if (octagon2.compareTo(octagon3) > 0)

   System.out.println("Octagon octagon2 has larger area than the Octagon octagon3");

 else

   System.out.println("Octagon octagon2 has smaller area than the Octagon octagon3");


System.out.println("Octagon octagon3: " + octagon3);

 System.out.println("-----------------------------------------------------------------");

EquilateralTriangle et1 = new EquilateralTriangle();
EquilateralTriangle et2 = new EquilateralTriangle(3.0);

System.out.println("Area of Equilateral Triangle et2 is " + et2.getArea());
System.out.println("Perimeter of Equilateral Triangle et2 is " + et2.getPerimeter());

if (et1.compareTo(et2) == 0)

   System.out.println("Equilateral Triangle et1 and et2 have equal coverage of area");

 else if (et1.compareTo(et2) > 0)
```

```
            System.out.println("Equilateral Triangle et1 has larger area than the Equilateral Triangle et2");

        else

            System.out.println("Equilateral Triangle et1 has smaller area than the Equilateral Triangle et2");


        EquilateralTriangle et3 = et2.clone();

        if (et2.compareTo(et3) == 0)

            System.out.println("Equilateral Triangle et2 and et3 have equal coverage of area");

         else if (et2.compareTo(et3) > 0)

            System.out.println("Equilateral Triangle et2 has larger area than the Equilateral Triangle et3");

        else

            System.out.println("Equilateral Triangle et2 has smaller area than the Equilateral Triangle et3");


        System.out.println("Equilateral Triangle et3: " + et3);

        System.out.println("------------------------------------------------------------------");

    }//main

}//Lab2GeometricShape
```

GeometricObject

```
package lab2geometricshape;


public abstract class GeometricObject {
    private String color = "not set";
    private boolean filled;
    private java.util.Date dateCreated;


    protected GeometricObject(){

        dateCreated = new java.util.Date();

    }//GeometricObject w/ no args

    protected GeometricObject(String color, boolean filled){

        this.color = color;
        this.filled = filled;
        dateCreated = new java.util.Date();

    }//GeometricObject w/ args

    public String toString(){

        return "Color: " + color + "\nIs filled: " + filled + "\nDate : " + dateCreated;

    }//toString

    public abstract double getArea();

    public abstract double getPerimeter();

}//GeometricObject
```

Circle

```
package lab2geometricshape;


public class Circle extends Ellipse implements Comparable, Cloneable{

    private double radius;

    public Circle(){

        super(1.0,1.0);
        radius = 1.0;

    }//circle w/ no args

    public Circle(double radius){

        super(radius,radius);
        System.out.println("This circle has a radius of: " + radius);
        this.radius = radius;

    }//circle w/ args

    public double getRadius(){

        return radius;

    }//getRadius

    public void setRadius(double radius){

        this.radius = radius;

    }//setRadius
```

```
public double getArea(){

   return radius * radius * Math.PI;

}//getArea

public double getPerimeter(){

   return 2 * radius * Math.PI;

}//getPerimeter

@Override
public String toString(){

   return "Cirle's radius: " + radius;

}//toString

@Override
public int compareTo(Object obj){

   if (this.getArea() > ((Circle) obj).getArea())

      return 1;

   else if (this.getArea() < ((Circle) obj).getArea())

      return -1;

   else

      return 0;

}//compareTo

@Override
public boolean equals(Object obj){

   return this.radius == ((Circle) obj).radius;

}//equals
```

```
    public Circle clone(){

        System.out.println("Getting Circle to clone...");

        return (Circle)super.clone();
    }//clone

}//Circle
```

Ellipse

```
package lab2geometricshape;


public class Ellipse extends GeometricObject implements Eccentric, Comparable, Cloneable {

    double a;
    double b;

    public Ellipse(){

        this.a = a;
        this.b = b;

    }//ellipse w/ no args

    public Ellipse(double side1, double side2){

        if(side1 < side2){

            a = side2;
            b = side1;

        }//if

        else{

            a = side1;
            b = side2;

        }//else
    }//ellipse w/ args
```

```
@Override
public double getPerimeter(){

   return (Math.PI) * (Math.sqrt(2 * (Math.pow(a,2) + Math.pow(b,2) + (a - b) / 2)));

}//getPerimeter

@Override
public double getArea(){

   return(Math.PI * a * b);

}//getArea

@Override
public double eccentricity(){

   double e = 0;
   e = Math.sqrt(a * a + b * b) / a;

   return e;
}//eccentricity

@Override
public String toString(){

   return "Ellipse Perimeter: " + getPerimeter() + "\nArea: " + getArea() + "\n";

}//toString

@Override
public int compareTo(Object obj){

   if(this.getArea() > ((Ellipse)obj).getArea())

      return  1;

   else if(this.getArea() < ((Ellipse)obj).getArea())

      return -1;

   else
```

```
            return 0;

    }//compareTo

    @Override
    public Ellipse clone(){
        try{
            System.out.print("Getting Ellipse to clone...");

            return(Ellipse) super.clone();

        }//try
        catch(Exception e){

            System.out.println("UH-OH in Ellipse");
            return null;

        }//catch

    }//clone
}//Ellipse
```

Equilateral Triangle

```
package lab2geometricshape;


public class EquilateralTriangle extends GeometricObject implements Comparable, Cloneable {

    double side;

    public EquilateralTriangle(){

        this.side = 1.0;

    }//EquilateralTriangle w/ no args

    public EquilateralTriangle(double a){

        side = a;

    }//EquilateralTriangle w/ args

    @Override
    public double getPerimeter(){

        return (side * 3);

    }//getPerimeter

    @Override
    public double getArea(){
```

```
    return ((side * side * Math.sqrt(3)) / 4 );

}//getArea

@Override
public String toString(){

    return "Equilateral Triangle Perimeter: " + getPerimeter() + "\nArea: " + getArea() + "\n";

}//toString

@Override
public int compareTo(Object obj) {

    if (this.getArea() > ((EquilateralTriangle) obj).getArea())

        return 1;

    else if (this.getArea() < ((EquilateralTriangle) obj).getArea())

        return -1;

    else

        return 0;

}//compareTo

@Override
public EquilateralTriangle clone(){
    try{
        System.out.print("Getting EquilateralTriangle to clone...");

        return(EquilateralTriangle) super.clone();

    }//try
    catch(Exception e){

        System.out.println("UH-OH in EquilateralTriangle");
        return null;

    }//catch
```

```
    }//clone

}//EQTriangle
```

Octagon

```
package lab2geometricshape;

public class Octagon extends GeometricObject implements Comparable, Cloneable {

    private double side;

    public Octagon(){

        this.side = 1.0;

    }//Octagon w/ no args

    public Octagon(double side){

        this.side = side;

    }//Octagon w/ args

    @Override
    public double getArea(){

        return (2 + 4 / Math.sqrt(2)) * side * side;

    }//getArea

    @Override
    public double getPerimeter(){
```

```
        return 8 * side;

    }//getPerimeter

    @Override
    public int compareTo(Object obj) {

        if (this.getArea() > ((Octagon) obj).getArea())

            return 1;

        else if (this.getArea() < ((Octagon) obj).getArea())

            return -1;

        else

            return 0;

    }//compareTo

     public Octagon clone(){
        try{
            System.out.print("Getting Ellipse to clone...");

            return(Octagon) super.clone();

        }//try
        catch(Exception e){

            System.out.println("UH-OH in Octagon");
            return null;

        }//catch

    }//clone


}//Octagon
```

Eccentric

```
package lab2geometricshape;


public interface Eccentric {

    double eccentricity();

}
```

Outputs

```
run:
This circle has a radius of: 3.0
Area of Circle circle2 is 28.274333882308138
Perimeter of Circle circle2 is 18.84955592153876
Circle circle1 has smaller area than the circle circle2
Getting Circle to clone...
Getting Ellipse to clone...Circle circle2 and circle3 have equal coverage of area
Circle circle3: Cirle's radius: 3.0
------------------------------------------------------------
Area of Ellipse ellipse2 is 28.274333882308138
Perimeter of Ellipse ellipse2 is 18.84955592153876
Ellipse ellipse1 has smaller area than the ellipse ellipse2
Getting Ellipse to clone...Ellipse ellipse2 and ellipse3 have equal coverage of area
Ellipse ellipse3: Ellipse Perimeter: 18.84955592153876
Area: 28.274333882308138

------------------------------------------------------------
Area of Octagon octagon2 is 43.45584412271571
Perimeter of Octagon octagon2 is 24.0
Octagon octagon1 has smaller area than the Octagon octagon2
Getting Ellipse to clone...Octagon octagon2 and octagon3 have equal coverage of area
Octagon octagon3: Color: not set
Is filled: false
```

Date : Thu Feb 13 18:47:05 EST 2020

Area of Equilateral Triangle et2 is 3.8971143170299736
Perimeter of Equilateral Triangle et2 is 9.0
Equilateral Triangle et1 has smaller area than the Equilateral Triangle et2
Getting EquilateralTriangle to clone...Equilateral Triangle et2 and et3 have equal coverage of area
Equilateral Triangle et3: Equilateral Triangle Perimeter: 9.0
Area: 3.8971143170299736

BUILD SUCCESSFUL (total time: 0 seconds)