Student Name: Tianle Shu

Student Id: 19232619

Lecturer Name: Seamus Hill

Question-1:

Part-A:

ShapesRelate Interface

```
//Student Name: Tianle Shu
//Student Id: 19232619
//Lecturer: Seamus Hill
package nuig.assignment7;

public interface ShapesRelate {
    public int compareShapes(ShapesRelate sr);

}
```

Shapes Class

```
//Student Name: Tianle Shu
//Student Id: 19232619
//Lecturer: Seamus Hill
package nuig.assignment7;

public abstract class Shapes implements ShapesRelate {
    protected double area;

    public Shapes() {

    }

    public double getArea(double area) {
        return area;
    }

    abstract void calculateArea();

    // this method from interface
    @Override
    public int compareShapes(ShapesRelate sr) {
        // let largest (object shapes) equal to sr which is from
attribute
```

```
        // casting let the Interface casting to the Object
        Shapes largest = (Shapes) sr;

        // if the Shapes's area > largest
        if (this.area > largest.area) {
            // then return 1
            return 1;
        } else {
            // else return 0
            return 0;
        }
    }
}
```

Circle Class

```
//Student Name: Tianle Shu
//Student Id: 19232619
//Lecturer: Seamus Hill

package nuig.assignment7;

public class Circle extends Shapes {
    private double radius;
    private double PI = 3.14;

    public Circle() {
        super();

    }

    public Circle(double radius) {
        this.radius = radius;
    }

    public void setRadius(double radius) {
        this.radius = radius;
    }

    public double getRadius() {
        return radius;
    }

    public void calculateArea() {
```

```
        area = PI*(radius*radius);
        System.out.println("The Circle area is : " + area);
    }

    @Override
    public String toString() {
        return "Circle [radius=" + radius + ", PI=" + PI + "]";
    }

}
```

Rectangle Class

```
//Student Name: Tianle Shu
//Student Id: 19232619
//Lecturer: Seamus Hill

package nuig.assignment7;

public class Rectangle extends Shapes {

    private double length;
    private double width;

    public Rectangle() {
        super();
    }

    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }

    // Method
    public void setWidth(double width) {
        this.width = width;

    }

    public double getWidth() {
        return width;
    }
```

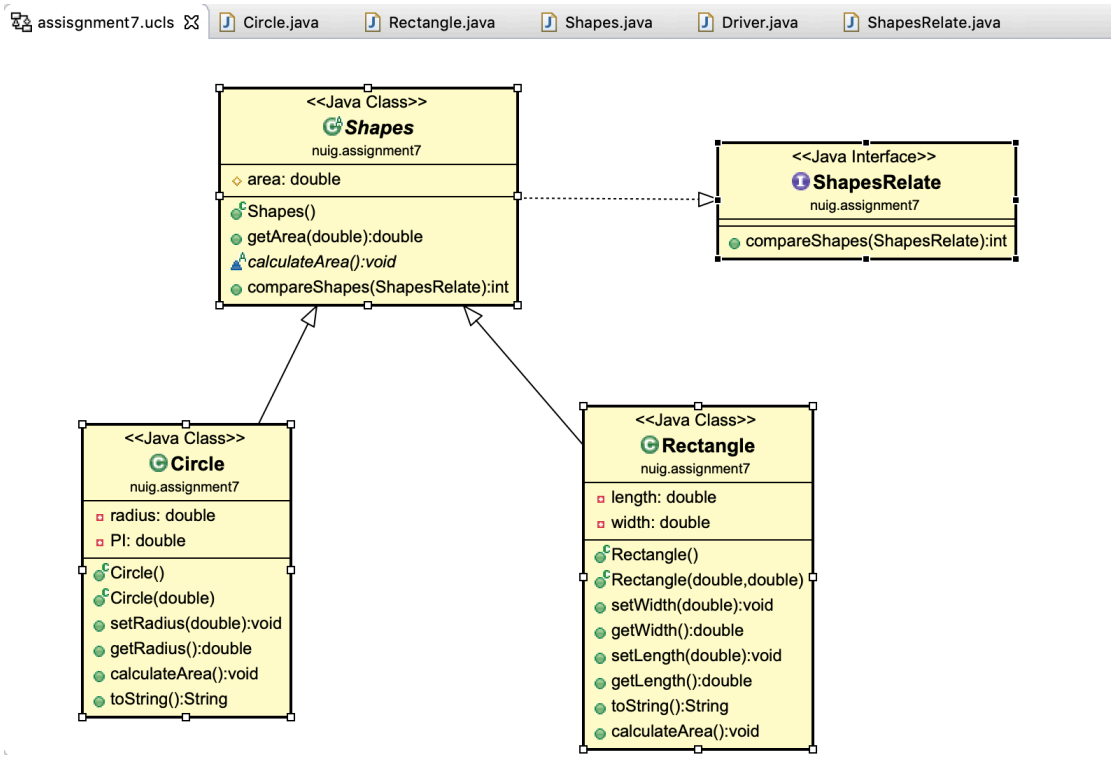
```
    public void setLength(double length) {
        this.length = length;
    }

    public double getLength() {
        return length;
    }

    // Override toString
    public String toString() {
        return "Rectangle [length=" + length + ", width=" +
width + "]";
        // return getLength();
    }

    public void calculateArea() {
        area = length * width;
        System.out.println("The Ractangle area is : " + area);
    }

}
```



Part-B:

Driver Class

```
//Student Name: Tianle Shu
//Student Id: 19232619
//Lecturer: Seamus Hill
package nuig.assignment7;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Driver {

   public static void main(String[] args) {

      // Scanner class, get user input, and it is found in the
java.util package.
      Scanner input = new Scanner(System.in);

      // Let people enter the number of the shapes to compare
      System.out.print("How many shapes you want to compare:
");
      // get the number from console
      int num = input.nextInt();
      // create a new ArrayList object, Shapes is element type
      // named shapes
      ArrayList<Shapes> shapes = new ArrayList<>(num);

      // use for loop
      // let the new object add to Arraylist shapes
      for (int i = 0; i < num; i++) {
         // let use choose the shapes
         System.out.print("\nenter 1 is circle\nenter 2 is
rectangle\nwhat kind of the shape? ");
         // get the shape from console
         int n = input.nextInt();

         // if user enter 1
         // then will have a circle object add into shapes
         if (n == 1) {
            System.out.print("please input the radius: ");
```

```
            // get the radius from console
            int radius = input.nextInt();
            // the circle object add into shapes
            shapes.add(new Circle(radius));
         }

         // if user enter 2
         // then will have a rectangle object add into shapes
         if (n == 2) {
            System.out.print("please input the width: ");
            // get the radius from width
            int width = input.nextInt();
            System.out.print("please input the length: ");
            // get the radius from length
            int length = input.nextInt();
            shapes.add(new Rectangle(width, length));
         }

      }//end for-loop

      // tell user output result
      System.out.println("\n--------Calculate result----------
-");
      // use for loop to print out the every shape's area.
      for (int i = 0; i < shapes.size(); i++) {
         // print out the every shape's area.
         // call the calculatearea() method.
         shapes.get(i).calculateArea();
      } //end for-loop

      // get the biggest area use the method (largestShape)
that will retuen an
      // object.
      System.out.println("\nThe largest area is : " +
largestShape(shapes).toString() + " area is " +
largestShape(shapes).area);
      // let the scanner close
      input.close();
   }// end main method

   // largestShape method to return an object which has the
bigget area
   public static Shapes largestShape(List<Shapes> list) {
      // assign first list as largest
```

```
        Shapes largest = list.get(0);
        // use foreach() for iterating through an array(list)
        for (Shapes shape : list) {
            // call compareShapes
            // compareShapes => check current is/not greater than
largest. if so, return 1,
            // otherwise 0
            if (shape.compareShapes(largest) == 1) {
                // if get return 1 then let largest = shape
                largest = shape;
                // output currently largest area
                // System.out.println("Currently of The largest: "
+ largest.area);
            }
        }
        // return the object which area is the largest
        return largest;
    }// end the largestShape method

}// end the class
```

```
Problems  Javadoc  Declaration  Search  Diagrams  Console
<terminated> Driver [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_161.jdk/Contents/Home/
How many shapes you want to compare: 3

enter 1 is circle
enter 2 is rectangle
what kind of the shape? 1
please input the radius: 9

enter 1 is circle
enter 2 is rectangle
what kind of the shape? 2
please input the width: 9
please input the length: 10

enter 1 is circle
enter 2 is rectangle
what kind of the shape? 1
please input the radius: 7

--------Calculate result-----------
The Circle area is : 254.34
The Ractangle area is : 90.0
The Circle area is : 153.86

The largest area is : Circle [radius=9.0, PI=3.14] area is 254.34
```