## Source code

### Shape.java:

```
package geometry;


public class Shape {
    public void displayArea() {
        System.out.println("This is the parent class.");
    }
}
```

### Rectangle.java:

```
package geometry;


public class Rectangle extends Shape {
    private double length;
    private double width;


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


    public void calculateArea() {
        double area = length * width;
        System.out.println("Area of Rectangle: " + area);
    }
}
```

### Circle.java:

```
package geometry;
```

```
public class Circle extends Shape {
    private double radius;

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

    public void calculateArea() {
        double area = Math.PI * Math.pow(radius, 2);
        System.out.println("Area of Circle: " + area);
    }
}
```

**Triangle.java:**

```
package geometry;

public class Triangle extends Shape {
    private double base;
    private double height;

    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
    }

    public void calculateArea() {
        double area = 0.5 * base * height;
        System.out.println("Area of Triangle: " + area);
    }
}
```

**GeometryMain.java:**

```
package geometry;

import java.util.ArrayList;

public class GeometryMain {

    public static void main(String[] args) {
        ArrayList<Shape> shapes = new ArrayList<>();

        shapes.add(new Rectangle(5, 10));
        shapes.add(new Circle(7));
        shapes.add(new Triangle(4, 6));

        try {
            for (Shape shape : shapes) {
                shape.displayArea();
                if (shape instanceof Rectangle) {
                    ((Rectangle) shape).calculateArea();
                } else if (shape instanceof Circle) {
                    ((Circle) shape).calculateArea();
                } else if (shape instanceof Triangle) {
                    ((Triangle) shape).calculateArea();
                }
                System.out.println("----------");
            }
        } catch (Exception e) {
            System.out.println("An error occurred: " + e.getMessage());
        } finally {
            System.out.println("Finally block executed.");
```

```
    }
  }
}
```