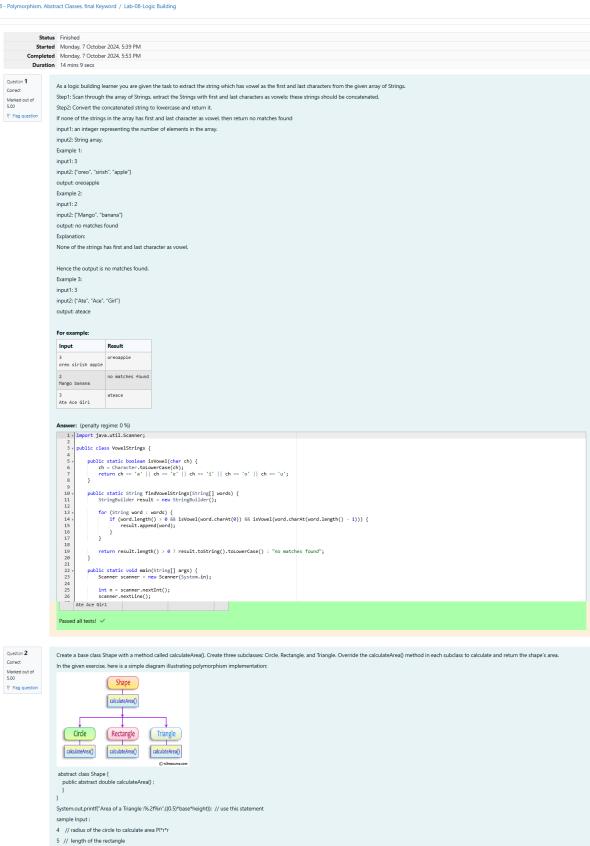
REC-CIS MITHESH THARUN S 2023-CSE-C M2 ~

## CS23333-Object Oriented Programming Using Java-2023

Dashboard / My courses / CS23333-OOPUJ-2023 / Lab-08 - Polymorphism, Abstract Classes, final Keyword / Lab-08-Logic Building





6 // breadth of the rectangle to calculate the area of a rectangle 4 // base of the triangle 3 // height of the triangle OUTPUT: calculateArea() calculateArea() abstract class Shape { public abstract double calculateArea();
}

```
System.out.printf("Area of a Triangle :%.2f%n",((0.5)*base*height)); // use this statement
sample Input :
4 // radius of the circle to calculate area PI*r*r
5 // length of the rectangle
6 // breadth of the rectangle to calculate the area of a rectangle
4 // base of the triangle
 Area of a circle :50.27
Area of a Rectangle :30.00
Area of a Triangle :6.00
 For example:
  Test Input Result
 1 4 Area of a circle: 50.27
5 Area of a Rectangle: 30.00
6 Area of a Triangle: 6.00
      7 Area of a circle: 153.94
4.5 Area of a Rectangle: 29.25
6.5 Area of a Triangle: 4.32
2.4
3.6
 1 v class FinalExample {
              final int maxSpeed = 120;
public final void displayMaxSpeed() {
    System.out.println("The maximum speed is: " + maxSpeed + " km/h");
   Test Expected
```

```
Test Expected Got

✓ 1 The maximum speed is: 120 km/h This is a subclass of FinalExample. This is a subclass of FinalExample.

Passed all tests! ✓
```

Finish review

```
- Lab-08-MCQ
                                                                                                                                              Jump to...
                                                                                                                                                                                                                                                                                                                                                                                                          FindStringCode -
                                                 double radius;
                                    Circle(double radius) {
   this.radius = radius;
                                                      @Override
public double calculateArea() {
   return Math.PI * radius * radius;
                                                class Rectangle extends Shape {
   double length, breadth;
                                                     Rectangle(double length, double breadth) {
   this.length = length;
   this.breadth = breadth;
}
                                                     @Override
public double calculateArea() {
    return length * breadth;
}
                                               class Triangle extends Shape {
   double base, height;
                                                    Triangle(double base, double height) {
    this.base = base;
    this.height = height;
}
                                                     @Override
public double calculateArea() {
    return 0.5 * base * height;
}
                                                public class Main {
   public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        double radius = scanner.nextDouble();
}
                                                abstract class Shape {
   public abstract double calculateArea();
                                   6 7 7 8 8 9 10 11 11 12 13 14 15 7 16 17 18 19 20 7 21 22 23 7 24 25 26 27 28 29 7 30 31
                                                      Circle(double radius) {
   this.radius = radius;
                                                    @Override
public double calculateArea() {
    return Math.PI * radius * radius;
}
                                                class Rectangle extends Shape {
   double length, breadth;
                                                     Rectangle(double length, double breadth) {
  this.length = length;
  this.breadth = breadth;
}
                                                      @Override
public double calculateArea() {
    return length * breadth;
```

```
### States Trigante ** States Tr
```