**PIJ Assignment 5**

Name: Samarth Bhadane

PRN: 24070126503

Batch: A2

**Code:**

// Main.java

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        // creating the menu

        while (true) {

            System.out.println("\nChoose a shape:");

            System.out.println("1. Circle");

            System.out.println("2. Rectangle");

            System.out.println("3. Square");

            System.out.println("4. Sphere");

            System.out.println("5. Cylinder");

            System.out.println("6. Equilateral Pyramid");

            System.out.println("7. Exit");

            System.out.print("Enter your choice: ");

            int choice = sc.nextInt();

            Shape shape = null;

            Volume volumeShape = null;

            switch (choice) {

                case 1:

                    System.out.print("Enter the radius of the circle: ");

                    double radius = sc.nextDouble();

                    shape = new Circle(radius);

                    break;

                case 2:

                    System.out.print("Enter the length of the rectangle: ");

                    double length = sc.nextDouble();

                    System.out.print("Enter the width of the rectangle: ");

                    double width = sc.nextDouble();

                    shape = new Rectangle(length, width);

                    break;

                case 3:

                    System.out.print("Enter the side of the square: ");

                    double side = sc.nextDouble();

                    shape = new Square(side);

                    break;

                case 4:

                    System.out.print("Enter the radius of the sphere: ");

                    double sphereRadius = sc.nextDouble();

                    volumeShape = new Sphere(sphereRadius);

                    break;

                case 5:

                    System.out.print("Enter the radius of the cylinder: ");

                    double cylinderRadius = sc.nextDouble();

                    System.out.print("Enter the height of the cylinder: ");

                    double height = sc.nextDouble();

                    volumeShape = new Cylinder(cylinderRadius, height);

                    break;

                case 6:

                    System.out.print("Enter the base side of the equilateral pyramid: ");

                    double baseSide = sc.nextDouble();

                    System.out.print("Enter the height of the equilateral pyramid: ");

                    double pyramidHeight = sc.nextDouble();

                    volumeShape = new EquilateralPyramid(baseSide, pyramidHeight);

                    break;

                case 7:

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

                    System.exit(0);

                    break;

                default:

                    System.out.println("Invalid choice");

                    continue;

            }

            if (shape != null) {

                System.out.println("Area: " + shape.calculateArea());

                System.out.println("Perimeter: " + shape.calculatePerimeter());

            }

            if (volumeShape != null) {

                if (volumeShape instanceof Shape) {

                    System.out.println("Surface Area: " + ((Shape) volumeShape).calculateArea());

                }

                System.out.println("Volume: " + volumeShape.calculateVolume());

            }

        }

    }

}

// Shape.java

// abstract class

public abstract class Shape {

    protected String shapeName;

    // constructor

    public Shape(String shapeName) {

        this.shapeName = shapeName;

    }

    // abstract method for calculating area and perimeter

    public abstract double calculateArea();

    public abstract double calculatePerimeter();

}

// Volume.java

// interface for 3D shapes requiring volume calculation

public interface Volume {

    double calculateVolume();

}

// Circle.java

import java.lang.Math;

// circle class extends shape class

public class Circle extends Shape {

    private double radius;

    // constructor

    public Circle(double radius) {

        super("Circle");

        this.radius = radius;

    }

    @Override

    public double calculateArea() {

        return Math.PI \* radius \* radius;

    }

    @Override

    public double calculatePerimeter() {

        return 2 \* Math.PI \* radius;

    }

}

// Rectangle.java

// rectangle class extends shape class

public class Rectangle extends Shape {

    private double length, width;

    // constructor

    public Rectangle(double length, double width) {

        super("Rectangle");

        this.length = length;

        this.width = width;

    }

    @Override

    public double calculateArea() {

        return length \* width;

    }

    @Override

    public double calculatePerimeter() {

        return 2 \* (length + width);

    }

}

//Square.java

// square class extends shape class

public class Square extends Shape {

    private double side;

    // constructor

    public Square(double side) {

        super("Square");

        this.side = side;

    }

    @Override

    public double calculateArea() {

        return side \* side;

    }

    @Override

    public double calculatePerimeter() {

        return 4 \* side;

    }

}

// Sphere.java

// sphere class extends shape class implements volume interface

public class Sphere extends Shape implements Volume {

    private double radius;

    // constructor

    public Sphere(double radius) {

        super("Sphere");

        this.radius = radius;

    }

    @Override

    public double calculateArea() {

        return 4 \* Math.PI \* radius \* radius;

    }

    @Override

    public double calculatePerimeter() {

        return 0; // as perimeter is not applicable for sphere

    }

    @Override

    public double calculateVolume() {

        return (4.0 / 3.0) \* Math.PI \* radius \* radius \* radius;

    }

}

// Cylinder.java

// cylinder class extends shape class implements volume interface

public class Cylinder extends Shape implements Volume {

    private double radius, height;

    // constructor

    public Cylinder(double radius, double height) {

        super("Cylinder");

        this.radius = radius;

        this.height = height;

    }

    @Override

    public double calculateArea() {

        return 2 \* Math.PI \* radius \* (radius + height);

    }

    @Override

    public double calculatePerimeter() {

        return 2 \* Math.PI \* radius;

    }

    @Override

    public double calculateVolume() {

        return Math.PI \* radius \* radius \* height;

    }

}

// EquilateralPyramid.java

// class EquilateralPyramid extends shape implements volume interface

public class EquilateralPyramid extends Shape implements Volume {

    private double baseSide, height;

    // constructor

    public EquilateralPyramid(double baseSide, double height) {

        super("Equilateral Pyramid");

        this.baseSide = baseSide;

        this.height = height;

    }

    @Override

    public double calculateArea() {

        double baseArea = baseSide \* baseSide;

        double slantHeight = Math.sqrt((baseSide / 2) \* (baseSide / 2) + height \* height);

        double lateralArea = 2 \* baseSide \* slantHeight;

        return baseArea + lateralArea;

    }

    @Override

    public double calculatePerimeter() {

        return 4 \* baseSide;

    }

    @Override

    public double calculateVolume() {

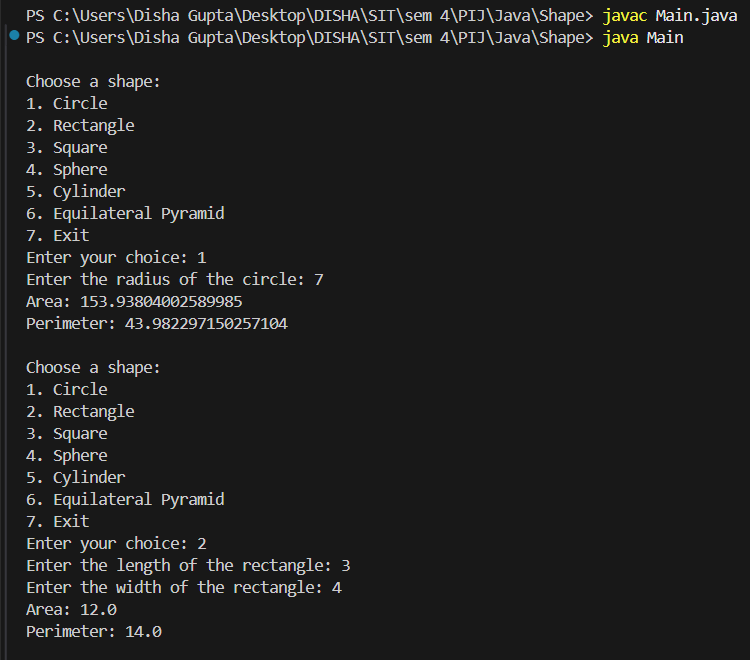
        return (1.0 / 3.0) \* (baseSide \* baseSide) \* height;

    }

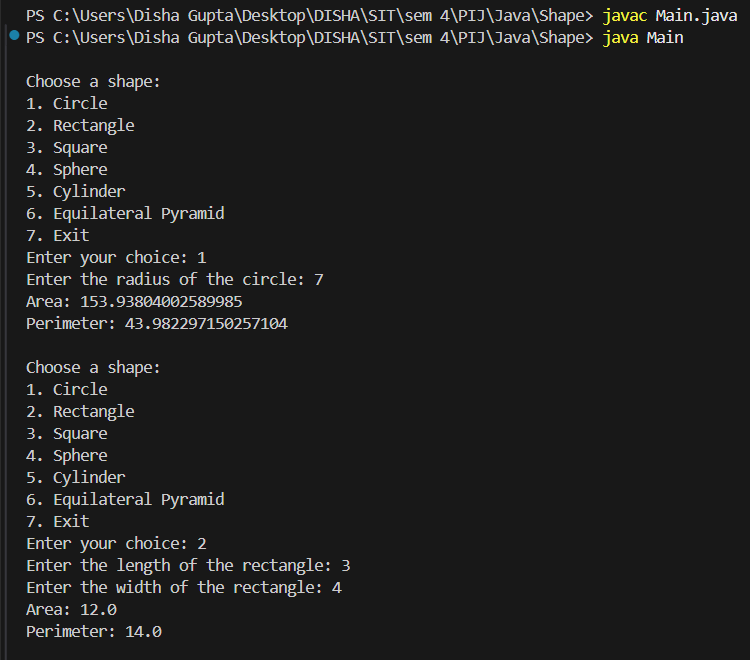
}

**Output:**

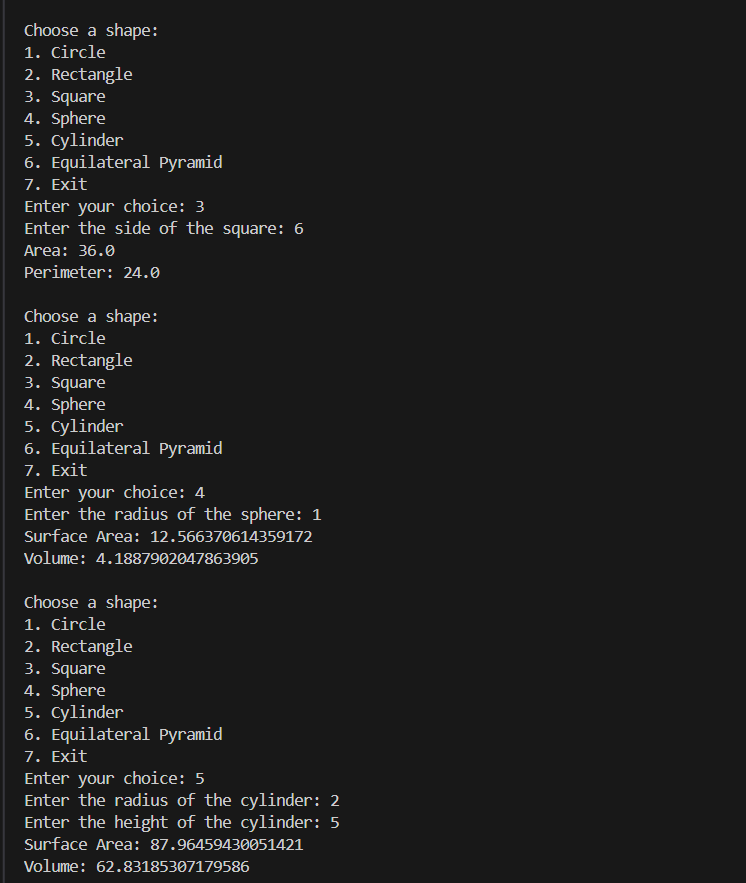
1. Circle



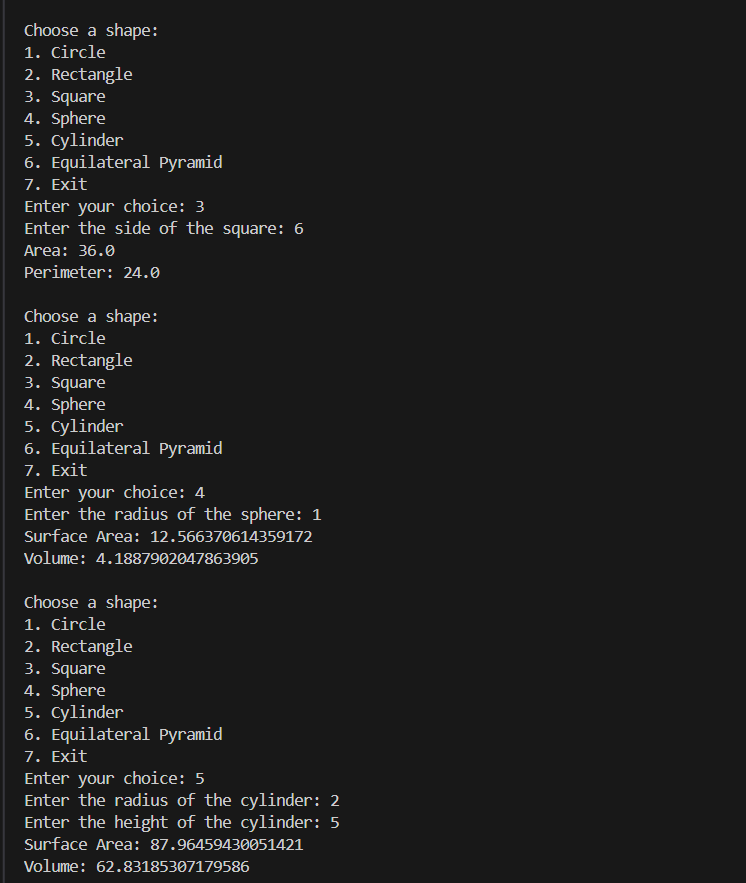
1. Rectangle



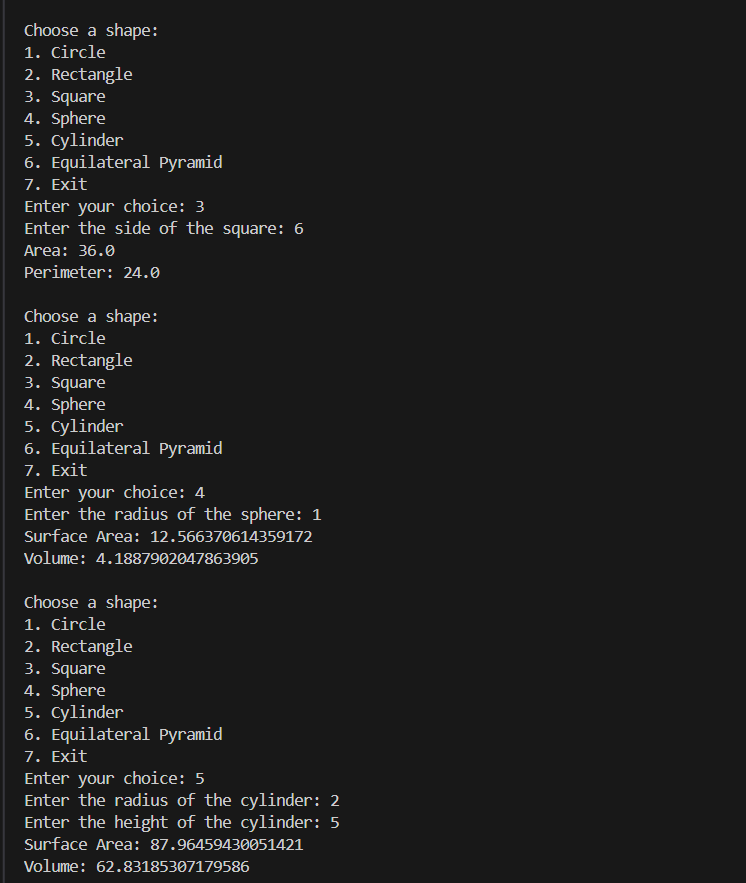
1. Square



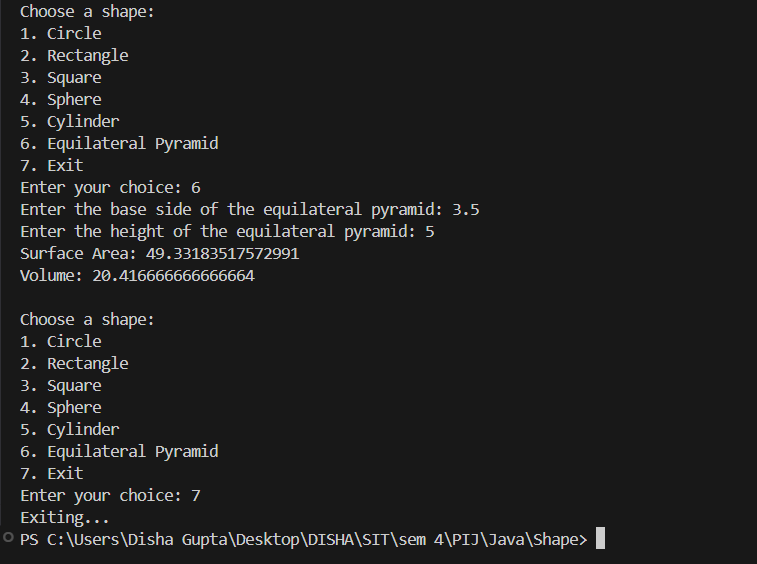
1. Sphere



1. Cylinder



1. Equilateral Pyramid



Link to the repository: <https://github.com/samarthsb4real/PIJ-Assignment-5>