```java
// Circle.java package
Assign5;
import
java.util.*;
 public class Circle extends Shape implements Volume
{     private double radius;

    @Override     public double
calculateShape() {
        return Math.PI * Math.pow(radius, 2);
    }

    @Override     public double
calculatePerimeter() {         return 2
* Math.PI * radius;
    }

    @Override     public double
calculateVolume() {
        // Volume calculation for a 3D circle (sphere) is not included
for simplicity         return 0;
    }

    // Function to get input from the user
public void getInput() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter   the   radius   of   the   circle:   ");
this.radius = scanner.nextDouble();
    }
}
```


```java
// Circle.java package
Assign5;
import
java.util.*;
 public class Circle extends Shape implements Volume
{     private double radius;

    @Override     public double
calculateShape() {
        return Math.PI * Math.pow(radius, 2);
    }
```

```
    @Override     public double
calculatePerimeter() {         return 2 *
Math.PI * radius;
    }

    @Override     public double
calculateVolume() {
        // Volume calculation for a 3D circle (sphere) is not included for
simplicity         return 0;
    }

    // Function to get input from the user
public void getInput() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter    the    radius    of    the    circle:    ");
this.radius = scanner.nextDouble();
    }
}
```

```
// Pyramid.java package
Assign5;
import java.util.Scanner;
 public class Pyramid extends Shape implements Volume
{     private double baseLength;     private double
baseWidth;     private double height;

    @Override
    public double calculateShape() {
// Surface area of a pyramid
        return baseLength * baseWidth + 0.5 * baseLength *
Math.sqrt(Math.pow(baseWidth / 2, 2) + Math.pow(height, 2))
                + 0.5 * baseWidth * Math.sqrt(Math.pow(baseLength / 2, 2) +
Math.pow(height, 2));
    }

    @Override     public double
calculatePerimeter() {
        // Perimeter calculation for a 3D shape is not applicable
return 0;
    }

    @Override
```

```
    public double calculateVolume() {          return (1.0 / 3.0) *
baseLength * baseWidth * height;      }

    // Function to get input from the user      public void
getInput() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the base length of the pyramid: ");
this.baseLength = scanner.nextDouble();
        System.out.print("Enter the base width of the pyramid: ");
this.baseWidth = scanner.nextDouble();
        System.out.print("Enter    the    height    of    the    pyramid:    ");
this.height = scanner.nextDouble();
    }
}
```

```
// Rectangle.java
package Assign5;
import java.util.Scanner;
 public class Rectangle extends Shape
{     private double length;
private double width;

    @Override     public double
calculateShape() {          return
length * width;
    }

    @Override
    public double calculatePerimeter() {
return 2 * (length + width);
    }

    // Function to get input from the user
public void getInput() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the length of the rectangle: ");
this.length = scanner.nextDouble();
        System.out.print("Enter the width of the rectangle: ");
this.width = scanner.nextDouble();
    }
}
```

```java
// Shape.java package
Assign5; public abstract
class Shape {     // Non-
abstract method
    public void showShape(String shape) {
        System.out.println("Selected shape: " + shape);
    }

    // Abstract methods to be implemented by
subclasses     public abstract double
calculateShape();     public abstract double
calculatePerimeter();
}
```

```java
// Main.java
package Assign5;
import java.util.Scanner;
 public class Main {     // Main function
public static void main(String[] args) {
        // Add your name, PRN, and Batch details in comments here
        Scanner scanner = new Scanner(System.in);

        // Menu-driven program
while (true) {
            System.out.println("\nSelect 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. Pyramid");
            System.out.println("0. Exit");
             int choice =
scanner.nextInt();
```

```
            switch (choice)
{               case 1:

calculateCircleAreaAndPerimeter();
break;                  case 2:

calculateRectangleAreaAndPerimeter();
break;                  case 3:

calculateSquareAreaAndPerimeter();
break;                  case 4:

calculateSphereAreaAndVolume();
break;                  case 5:

calculateCylinderAreaAndVolume();
break;                  case 6:

calculatePyramidAreaAndVolume();
break;                  case 0:
                    System.out.println("Ending program... So long
comrade!");                     System.exit(0);                 default:
                    System.out.println("Invalid choice. Please try again.");
            }
        }
    }

    // Function to calculate the area and perimeter of a
circle     private static void
calculateCircleAreaAndPerimeter() {         Circle circle =
new Circle();         circle.showShape("Circle");
circle.getInput();
         double area =
circle.calculateShape();
        double perimeter = circle.calculatePerimeter();
        System.out.println("Area: " + area);
        System.out.println("Perimeter: " + perimeter);
    }

    // Function to calculate the area and perimeter of a
rectangle     private static void
calculateRectangleAreaAndPerimeter() {         Rectangle
rectangle = new Rectangle();
rectangle.showShape("Rectangle");         rectangle.getInput();
```

```
        double area = rectangle.calculateShape();
        double perimeter = rectangle.calculatePerimeter();
        System.out.println("Area: " + area);
        System.out.println("Perimeter: " + perimeter);
    }

    // Function to calculate the area and perimeter of a
square     private static void
calculateSquareAreaAndPerimeter() {         Square square =
new Square();         square.showShape("Square");
square.getInput();
         double area =
square.calculateShape();
        double perimeter = square.calculatePerimeter();
        System.out.println("Area: " + area);
        System.out.println("Perimeter: " + perimeter);
    }

    // Function to calculate the area and volume of a
sphere     private static void
calculateSphereAreaAndVolume() {         Sphere sphere =
new Sphere();         sphere.showShape("Sphere");
sphere.getInput();
         double area = sphere.calculateShape();
double volume = sphere.calculateVolume();

        System.out.println("Surface Area: " + area);
        System.out.println("Volume: " + volume);
    }

    // Function to calculate the area and volume of a
cylinder     private static void
calculateCylinderAreaAndVolume() {         Cylinder cylinder
= new Cylinder();         cylinder.showShape("Cylinder");
cylinder.getInput();
         double area = cylinder.calculateShape();
double volume = cylinder.calculateVolume();

        System.out.println("Surface Area: " + area);
        System.out.println("Volume: " + volume);
    }

    // Function to calculate the area and volume of a pyramid
private static void calculatePyramidAreaAndVolume() {
```

```
        Pyramid pyramid = new Pyramid();
pyramid.showShape("Pyramid");          pyramid.getInput();
         double area = pyramid.calculateShape();
double volume = pyramid.calculateVolume();

        System.out.println("Surface Area: " + area);
        System.out.println("Volume: " + volume);
    }
}
```

Output:

```
Select a shape:
1. Circle
2. Rectangle
3. Square
4. Sphere
5. Cylinder
6. Pyramid
0. Exit
1
Selected shape: Circle
Enter the radius of the circle: 6
Area: 113.09733552923255
Perimeter: 37.69911184307752

Select a shape:
1. Circle
2. Rectangle
3. Square
4. Sphere
5. Cylinder
6. Pyramid
0. Exit
3
Selected shape: Square
Enter the side length of the square: 10
Area: 100.0
Perimeter: 40.0
```

Github : https://github.com/RajhansMore/Codes.logic