****

**Name**: ALI ABID

**Student ID**: BSCS-KC-006

**Semester**: 2.

**Course Name**: Object Oriented Programming-Lab.

**Course Code**: CSOO122L.

**Teacher**: Syed Zubair Ali.

**National University of Modern Languages**

**Karachi Campus**

**Faculty of Engineering & Computing**

**Department of Computer Science**

LAB 1

**Explanation:**

Convert temperature between Centigrade and Fahrenheit using a formula. Implement with and without user input. Display name and roll number.

**Code:**

import java.util.Scanner;

public class TemperatureConverter {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Method 1: Without user interaction (hardcoded value)

        double celsius1 = 25.0; // Example hardcoded value

        double fahrenheit1 = celsius1 \* 9 / 5 + 32;

        System.out.println("\nWithout user interaction:");

        System.out.printf("Celsius: %.2f°C -> Fahrenheit: %.2f°F%n%n", celsius1, fahrenheit1);

        // Method 2: With user interaction

        Scanner scanner = new Scanner(System.in);

        System.out.println("With user interaction:");

        System.out.print("Enter temperature in Celsius: ");

        double celsius2 = scanner.nextDouble();

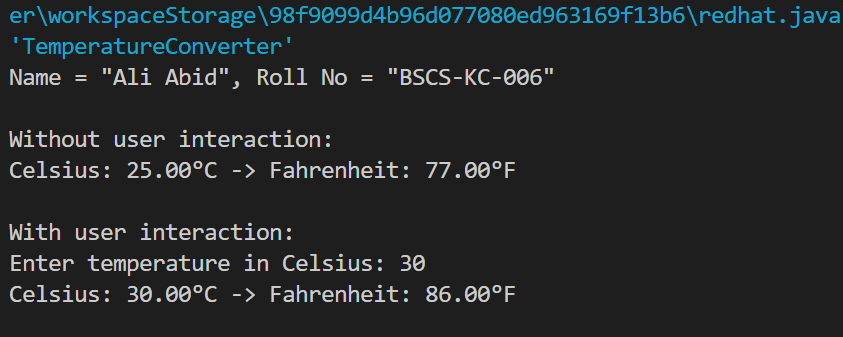
        double fahrenheit2 = celsius2 \* 9 / 5 + 32;

        System.out.printf("Celsius: %.2f°C -> Fahrenheit: %.2f°F%n%n", celsius2, fahrenheit2);

    }

}

**Output:**



LAB 2

**Explanation:**

Check if a user-entered amount can be withdrawn based on the current balance. Print appropriate messages for available or unavailable funds.

**Code:**

import java.util.Scanner;

public class BalanceChecker {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Define the current balance

        double currentBalance = 5000.00; // Example balance

        // Take input

        Scanner scanner = new Scanner(System.in);

        System.out.print("\nEnter the amount to be drawn: ");

        double amount = scanner.nextDouble();

        // Check if the amount is available

        if (amount <= currentBalance) {

            System.out.println("Amount is available.");

        } else {

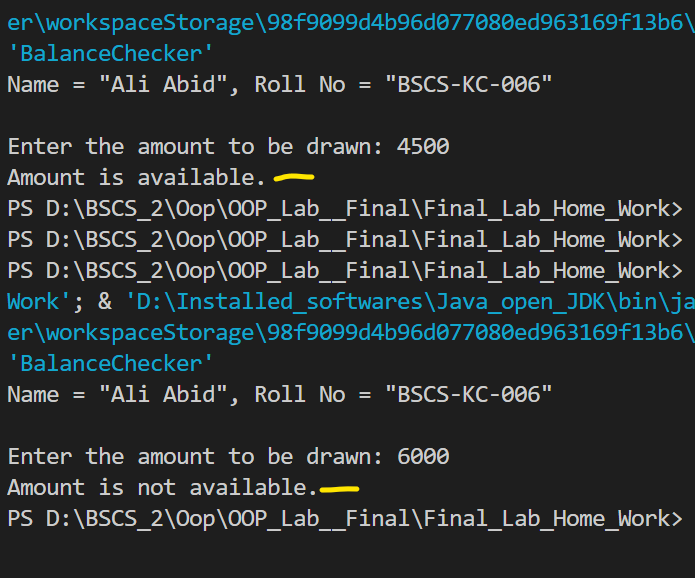
            System.out.println("Amount is not available.");

        }

    }

}

**Output:**



LAB 3

**Explanation:**

Use a while loop to prompt the user to select the correct answer from a list of choices for a question.

**Code:**

import java.util.Scanner;

public class QuizProgram {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        Scanner scanner = new Scanner(System.in);

        int userChoice = -1;

        // Display the question and answer choices

        System.out.println("\nQuestion: What is the capital of Pakistan?");

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

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

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

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

        // Correct answer

        int correctAnswer = 2;

        // While loop to prompt the user until they choose the correct answer

        while (userChoice != correctAnswer) {

            System.out.print("\nEnter your choice (1-4): ");

            userChoice = scanner.nextInt();

            if (userChoice == correctAnswer) {

                System.out.println("Correct! The capital of Pakistan is Islamabad.");

            } else {

                System.out.println("Incorrect! Please try again.");

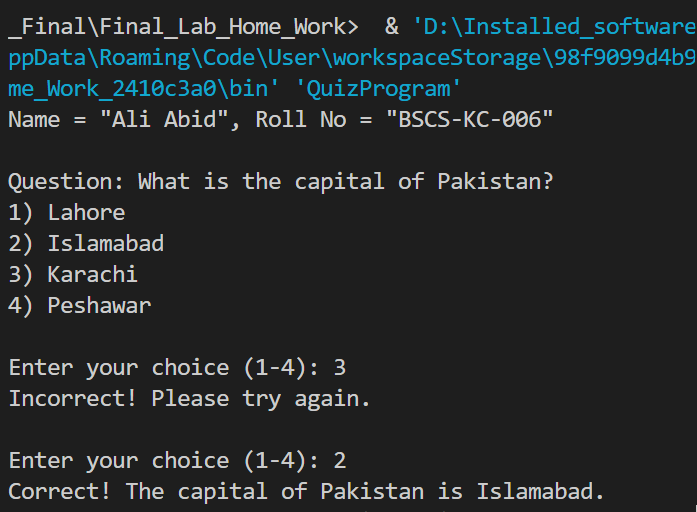
            }

        }

    }

}

**Output:**



LAB 4

**Explanation:**

Create a class MyMath with a method to round numbers to the nearest hundredth. Use it in another class TryMyMath to calculate and display a number, its square, and cube.

**Code:**

public class MyMath {

    // Static method to round a double to the nearest 100th

    public static double roundToNearest100th(double number) {

        return Math.round(number \* 100) / 100.0;

    }

}

public class TryMyMath {

    // Method to print a number, its square, and its cube (all rounded to 100th)

    public static void printNumberDetails(double number) {

        double roundedNumber = MyMath.roundToNearest100th(number);

        double roundedSquare = MyMath.roundToNearest100th(Math.pow(number, 2));

        double roundedCube = MyMath.roundToNearest100th(Math.pow(number, 3));

        System.out.println("Number (rounded to 100th): " + roundedNumber);

        System.out.println("Square (rounded to 100th): " + roundedSquare);

        System.out.println("Cube (rounded to 100th): " + roundedCube);

    }

    // Main method

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Call printNumberDetails with Math.E

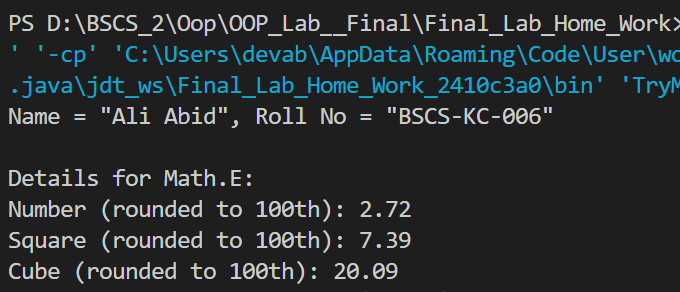
        System.out.println("\nDetails for Math.E:");

        printNumberDetails(Math.E);

    }

}

**Output:**



LAB 5

**Explanation:**

Simulate a bank account with features like deposits, withdrawals, interest rate settings, and balance checking. Demonstrate the account in action.

**Code:**

public class BankAccount {

    private double balance;

    private double interstate; // in percentage (e.g., 5 for 5%)

    // Constructor to initialize the account with a starting balance

    public BankAccount(double initialBalance) {

        this.balance = initialBalance;

        this.interestRate = 0; // Default interest rate

    }

    // Method to deposit money

    public void deposit(double amount) {

        if (amount > 0) {

            balance += amount;

            System.out.println("Deposited: " + amount);

        } else {

            System.out.println("Invalid deposit amount.");

        }

    }

    // Method to withdraw money

    public void withdraw(double amount) {

        if (amount > 0 && amount <= balance) {

            balance -= amount;

            System.out.println("Withdrawn: " + amount);

        } else if (amount > balance) {

            System.out.println("Insufficient balance.");

        } else {

            System.out.println("Invalid withdrawal amount.");

        }

    }

    // Method to set the interest rate

    public void setInterestRate(double rate) {

        if (rate >= 0) {

            this.interestRate = rate;

            System.out.println("Interest rate set to: " + rate + "%");

        } else {

            System.out.println("Invalid interest rate.");

        }

    }

    // Method to add interest to the balance

    public void addInterest() {

        double interest = balance \* (interestRate / 100);

        balance += interest;

        System.out.println("Interest added: " + interest);

    }

    // Method to get the current balance

    public double getBalance() {

        return balance;

    }

}

public class Main {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Create a BankAccount object with an initial balance

        BankAccount myAccount = new BankAccount(5000); // Initial balance: 5000

        // Perform operations on the account

        System.out.println("\nInitial Balance: " + myAccount.getBalance());

        myAccount.deposit(1500); // Deposit 1500

        myAccount.withdraw(2000); // Withdraw 2000

        myAccount.setInterestRate(5); // Set interest rate to 5%

        myAccount.addInterest(); // Add interest

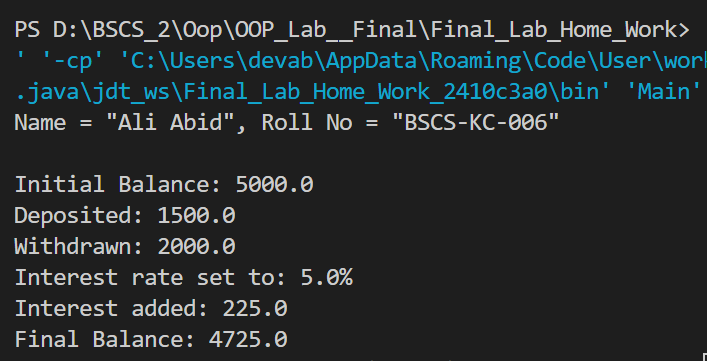
        // Display the final balance

        System.out.println("Final Balance: " + myAccount.getBalance());

    }

}

**Output:**



LAB 6

**Explanation:**

Create a Calculation class with methods for addition and subtraction. Take user input for variables and pass them to the methods for calculation.

**Code:**

import java.util.Scanner;

public class Calculation {

    private double a;

    private double b;

    // Method to perform addition

    public double Add() {

        return a + b;

    }

    // Method to perform subtraction

    public double Subtract() {

        return a - b;

    }

    // Method to take user input for a and b

    public void takeInput() {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the value for a: ");

        a = scanner.nextDouble();

        System.out.print("Enter the value for b: ");

        b = scanner.nextDouble();

    }

}

public class Main {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Create an object of Calculation

        Calculation calc = new Calculation();

        // Take user input for a and b

        calc.takeInput();

        // Perform addition and subtraction and display the results

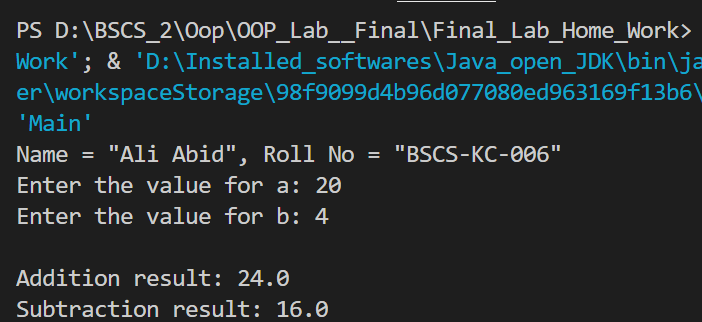
        System.out.println("\nAddition result: " + calc.Add());

        System.out.println("Subtraction result: " + calc.Subtract());

    }

}

**Output:**



LAB 7

**Explanation:**

Use multilevel inheritance to create a program for generating a marksheet, calculating percentages and averages for different departments.

**Code:**

public class Student {

    String name;

    int rollNo;

    // Constructor to initialize student details

    public Student(String name, int rollNo) {

        this.name = name;

        this.rollNo = rollNo;

    }

    // Method to display student details

    public void displayStudentDetails() {

        System.out.println("Student Name: " + name);

        System.out.println("Roll Number: " + rollNo);

    }

}

public class Department extends Student {

    String departmentName;

    // Constructor to initialize department details

    public Department(String name, int rollNo, String departmentName) {

        super(name, rollNo);

        this.departmentName = departmentName;

    }

    // Method to display department details

    public void displayDepartmentDetails() {

        System.out.println("Department: " + departmentName);

    }

}

public class Marks extends Department {

    double subject1, subject2, subject3, subject4, subject5;

    double totalMarks = 500; // Assuming total marks for 5 subjects is 500

    // Constructor to initialize marks

    public Marks(String name, int rollNo, String departmentName, double subject1, double subject2, double subject3, double subject4, double subject5) {

        super(name, rollNo, departmentName);

        this.subject1 = subject1;

        this.subject2 = subject2;

        this.subject3 = subject3;

        this.subject4 = subject4;

        this.subject5 = subject5;

    }

    // Method to calculate total marks

    public double calculateTotal() {

        return subject1 + subject2 + subject3 + subject4 + subject5;

    }

    // Method to calculate percentage

    public double calculatePercentage() {

        return (calculateTotal() / totalMarks) \* 100;

    }

    // Method to calculate average

    public double calculateAverage() {

        return calculateTotal() / 5;

    }

    // Method to display marksheet and calculations

    public void displayMarksheet() {

        displayStudentDetails();

        displayDepartmentDetails();

        System.out.println("Marks in Subject 1: " + subject1);

        System.out.println("Marks in Subject 2: " + subject2);

        System.out.println("Marks in Subject 3: " + subject3);

        System.out.println("Marks in Subject 4: " + subject4);

        System.out.println("Marks in Subject 5: " + subject5);

        System.out.println("Total Marks: " + calculateTotal() + " / " + totalMarks);

        System.out.println("Average: " + calculateAverage());

        System.out.println("Percentage: " + calculatePercentage() + "%");

    }

}

public class Main {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Create Marks object and initialize with student, department, and subject marks

        Marks studentMarks = new Marks("Ali Abid", 12345, "Computer Science", 85, 90, 88, 92, 87);

        // Display marksheet

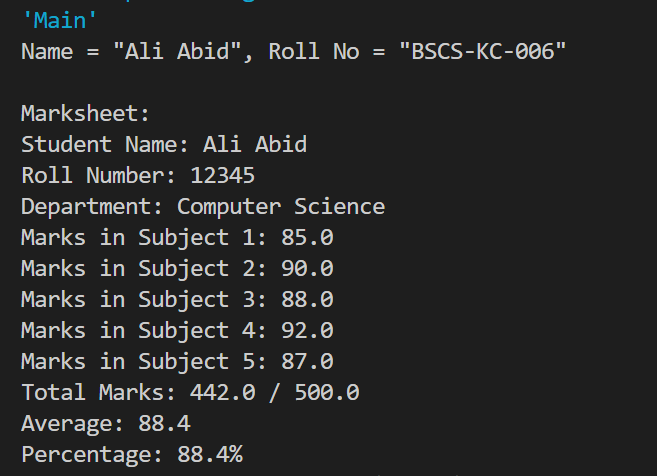
        System.out.println("\nMarksheet:");

        studentMarks.displayMarksheet();

    }

}

**Output:**



LAB 8

**Explanation:**

Create a Vehicle class with data members, constructors, and methods. Inherit Car and Bike classes, and override a virtual method to calculate speed.

**Code:**

abstract class Vehicle {

    String model;

    String color;

    String engine;

    int door;

    // Default Constructor

    Vehicle() {

        this.model = "Unknown";

        this.color = "Unknown";

        this.engine = "Unknown";

        this.door = 0;

    }

    // Parameterized Constructor

    Vehicle(String model, String color, String engine, int door) {

        this.model = model;

        this.color = color;

        this.engine = engine;

        this.door = door;

    }

    // Method to start the vehicle

    void start() {

        System.out.println(model + " is starting...");

    }

    // Method to stop the vehicle

    void stop() {

        System.out.println(model + " is stopping...");

    }

    // Method to run the vehicle

    void run() {

        System.out.println(model + " is running...");

    }

    // Abstract method to calculate speed

    abstract void calculateSpeed();

    // Method to print vehicle information

    void printInfo() {

        System.out.println("Model: " + model);

        System.out.println("Color: " + color);

        System.out.println("Engine: " + engine);

        System.out.println("Number of Doors: " + door);

    }

}

class Car extends Vehicle {

    // Constructor

    Car(String model, String color, String engine, int door) {

        super(model, color, engine, door);

    }

    // Override the calculateSpeed method

    @Override

    void calculateSpeed() {

        System.out.println("The car's average speed is 120 km/h.");

    }

}

class Bike extends Vehicle {

    // Constructor

    Bike(String model, String color, String engine, int door) {

        super(model, color, engine, door);

    }

    // Override the calculateSpeed method

    @Override

    void calculateSpeed() {

        System.out.println("The bike's average speed is 80 km/h.");

    }

}

public class Main {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Create a Car object

        Car car = new Car("Toyota Corolla", "White", "1.8L", 4);

        System.out.println("\nCar Information:");

        car.printInfo();

        car.start();

        car.run();

        car.calculateSpeed();

        car.stop();

        // Create a Bike object

        Bike bike = new Bike("Honda CB150F", "Red", "150cc", 0);

        System.out.println("\nBike Information:");

        bike.printInfo();

        bike.start();

        bike.run();

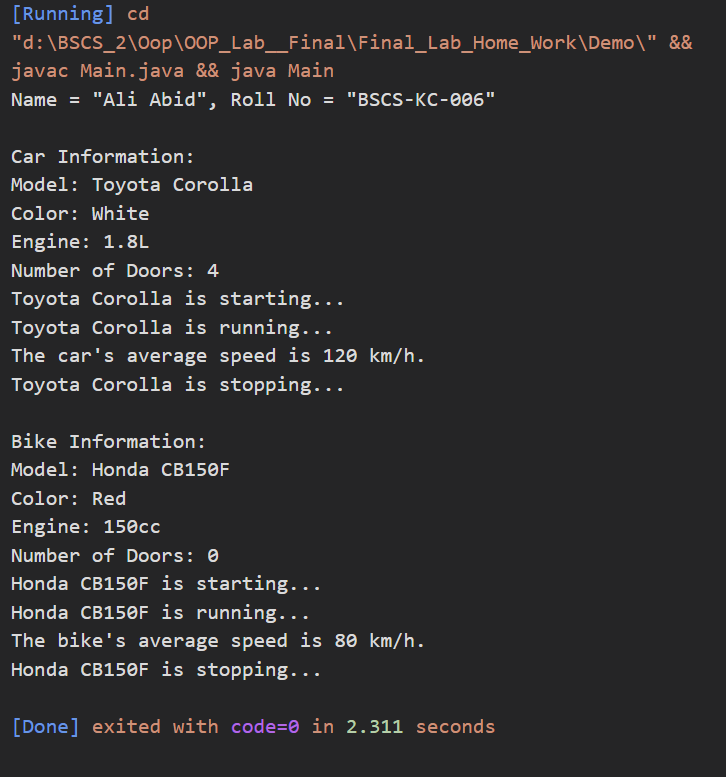
        bike.calculateSpeed();

        bike.stop();

    }

}

**Output:**



LAB 9

**Explanation:**

Write a program to print the elements of a string array using a function. Initialize the array in the main method and pass it to the function.

**Code:**

public class Main {

    // Method to print array elements

    static void printArray(String[] array) {

        System.out.println("Array Elements:");

        for (String element : array) {

            System.out.println(element);

        }

    }

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Initialize a string array

        String[] stringArray = {"Java", "Python", "C++", "JavaScript", "Django"};

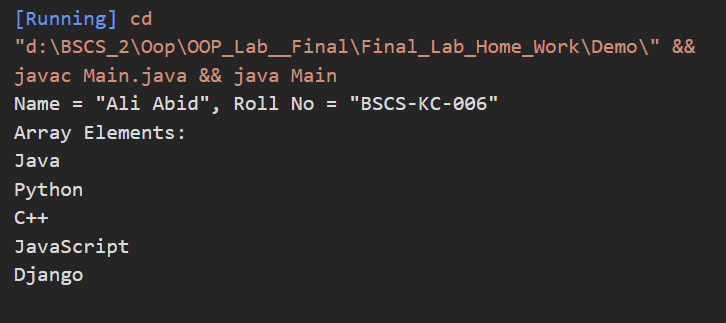
        // Call the printArray method and pass the array

        printArray(stringArray);

    }

}

**Output:**



LAB 10

**Explanation:**

Implement an interface IShape with methods to input a shape name and calculate its area. Use if-else to handle different shapes like Circle, Rectangle, and Triangle.

**Code:**

import java.util.Scanner;

interface IShape {

    void SName(); // Method to input the name of the shape

    void ShArea(); // Method to calculate and display the area of the shape

}

class ShapeCalculator implements IShape {

    private String shapeName;

    // Method to input the name of the shape

    @Override

    public void SName() {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the name of the shape (Circle, Rectangle, Triangle): ");

        shapeName = scanner.nextLine();

    }

    // Method to calculate and display the area of the shape

    @Override

    public void ShArea() {

        Scanner scanner = new Scanner(System.in);

        double area = 0;

        if (shapeName.equalsIgnoreCase("Circle")) {

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

            double radius = scanner.nextDouble();

            area = Math.PI \* radius \* radius;

            System.out.printf("The area of the Circle is: %.2f\n", area);

        } else if (shapeName.equalsIgnoreCase("Rectangle")) {

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

            double length = scanner.nextDouble();

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

            double width = scanner.nextDouble();

            area = length \* width;

            System.out.printf("The area of the Rectangle is: %.2f\n", area);

        } else if (shapeName.equalsIgnoreCase("Triangle")) {

            System.out.print("Enter the base of the triangle: ");

            double base = scanner.nextDouble();

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

            double height = scanner.nextDouble();

            area = 0.5 \* base \* height;

            System.out.printf("The area of the Triangle is: %.2f\n", area);

        } else {

            System.out.println("Invalid shape name entered!");

        }

    }

}

public class Main {

    public static void main(String[] args) {

        // Print name and roll number (common statement)

        System.out.println("Name = \"Ali Abid\", Roll No = \"BSCS-KC-006\"");

        // Create an object of ShapeCalculator

        ShapeCalculator shapeCalculator = new ShapeCalculator();

        // Input shape name and calculate area

        shapeCalculator.SName();

        shapeCalculator.ShArea();

    }

}

**Output:**

