

TASK4

1.Source code

```
import java.util.Scanner;

/**
 * Task 4: Method Design & Modular Calculator
 */

public class ModularCalculator {

    // Addition method (int)

    static int add(int a, int b) {

        return a + b;

    }

    // Addition method (double) - Method Overloading

    static double add(double a, double b) {

        return a + b;

    }

    // Subtraction

    static double subtract(double a, double b) {

        return a - b;

    }

    // Multiplication

    static double multiply(double a, double b) {

        return a * b;

    }

    // Division with exception handling

    static double divide(double a, double b) {
```

```

    if (b == 0) {
        throw new ArithmeticException("Division by zero is not allowed");
    }

    return a / b;
}

// Utility method to display result

static void displayResult(String operation, double result) {
    System.out.println("Result of " + operation + ": " + result);
}

// Method to demonstrate pass-by-value

static void changeValue(int x) {
    x = 100;

    System.out.println("Inside method, value is: " + x);
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    System.out.println("=== Modular Calculator ===");

    System.out.print("Enter first number: ");

    double num1 = sc.nextDouble();

    System.out.print("Enter second number: ");

    double num2 = sc.nextDouble();

    System.out.print("Choose operation (+ - * /): ");

    char choice = sc.next().charAt(0);

    try {
        switch (choice) {

```

```

    case '+':
        displayResult("Addition", add(num1, num2));

        break;

    case '-':
        displayResult("Subtraction", subtract(num1, num2));

        break;

    case '*':
        displayResult("Multiplication", multiply(num1, num2));

        break;

    case '/':
        displayResult("Division", divide(num1, num2));

        break;

    default:

        System.out.println("Invalid operation selected.");

    }
} catch (ArithmeticException e) {

    System.out.println("Error: " + e.getMessage());

}

// Pass-by-value demonstration

int value = 50;

System.out.println("\nBefore method call, value is: " + value);

changeValue(value);

System.out.println("After method call, value is: " + value);

sc.close();

}

```

}

2.Output:

```
=== Modular Calculator ===  
Enter first number: 10  
Enter second number: 2  
Choose operation (+ - * /): /  
Result of Division: 5.0  
  
Before method call, value is: 50  
Inside method, value is: 100  
After method call, value is: 50
```