Task 1: Data Types/Variables :

Write a program that declares two integer variables, swaps their values without using a third variable, and prints the result.

```
public class task1 {
    public static void main(String[] args) {
        // Declare two integer variables
        int num1 = 5, num2 = 10;

        // Print the initial values of the variables
        System.out.println("Before swapping: num1 = " + num1 + ", num2 = " + num2);

        // Swap the values of the variables without using a third variable
        num1 = num1 + num2; // num1 now holds the sum of the original values
        num2 = num1 - num2; // num2 now holds the difference between the sum and the original value of num2
        num1 = num1 - num2; // num1 now holds the difference between the sum and the original value of num1

        // Print the swapped values of the variables
        System.out.println("After swapping: num1 = " + num1 + ", num2 = " + num2);
    }
}
```

Output:

```
Before swapping: num1 = 5, num2 = 10
After swapping: num1 = 10, num2 = 5
```

Task 2: Operators

Create a program that simulates a simple calculator using command-line arguments to perform and print the result of addition, subtraction, multiplication, and division..

```
public class task2 {


    public static void main(String[] args) {
        int x=(5);
        int y=(4);


        int result;
        result=x+y;
        System.out.println("sum of two numbers:"+ result);
        result=x-y;
        System.out.println("subtraction of two integer:" + result);
        result=x*y;
        System.out.println("multiplication of two integer:" + result);
        result=x/y;
        System.out.println("division of two integer:" + result);


    }
}
```

Output:

```
sum of two numbers:9
subtraction of two integer:1
multiplication of two integer:20
division of two integer:1
```

Task 3: Control Flow

Write a Java program that reads an integer and prints whether it is a prime number using a for loop and if statements.

```
import java.util.Scanner;


public class task3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);


        System.out.print("Enter an integer: ");
        int num = scanner.nextInt();


        if (num < 2) {
            System.out.println(num + " is not a prime number.");
            return;
        }


        boolean isPrime = true;


        for (int i = 2; i <= num / 2; i++) {
            if (num % i == 0) {
                isPrime = false;
                break;
            }
        }


        if (isPrime) {
    }

            System.out.println(num + " is a prime number.");

        } else {

            System.out.println(num + " is not a prime number.");

        }

    }

}
```

Output1:

```
Enter an integer: 17

17 is a prime number.
```

Task 4: Constructors

Implement a Matrix class that has a constructor which initializes the dimensions of a matrix and a method to fill the matrix with values.

```
public class task4 {

    private int[][] data;

    private int rows;

    private int cols;



    public task4(int rows, int cols) {

        this.rows = rows;

        this.cols = cols;

        this.data = new int[rows][cols];

    }



    public void fill(int[] values) {
        if (values.length != rows * cols) {
            throw new IllegalArgumentException("Number of values must match
matrix dimensions");
        }
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                this.data[i][j] = values[i * cols + j];
            }
        }
    }

    public void display() {
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                System.out.print(this.data[i][j] + " ");
            }
System.out.println();
        }
    }

    public static void main(String[] args) {
        Matrix matrix = new Matrix(3, 2);
        int[] values = {1, 2, 3, 4, 5, 6};
        matrix.fill(values);
        matrix.display();
    }
}
```

Output:

```
1 2
3 4
5 6
```

Task 5: Inheritance

Create a Shape class with a method area() and extend it with Circle and Rectangle classes overriding the area() method appropriately.

```
class Shape {
    public double area() {
        return 0.0; // Default implementation for unknown shape
    }
}

class Circle extends Shape {
    private double radius;

    public Circle(double radius) {
        this.radius = radius;
    }
```

```
}


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

class Rectangle extends Shape {
    private double length;
    private double width;

    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }

    @Override
    public double area() {
        return length * width;
    }
}
public class task5 {
    public static void main(String[] args) {
        Shape circle = new Circle(5.0);
        Shape rectangle = new Rectangle(4.0, 3.0);

        System.out.println("Area of Circle: " + circle.area());
        System.out.println("Area of Rectangle: " + rectangle.area());
    }
}
```

Output:

```
Area of Circle: 78.53981633974483
Area of Rectangle: 12.0
```