# Rajalakshmi Engineering College

Name: Sivaranjanii M
Email: 241901109@rajalakshmi.edu.in
Roll no: 241901109
Phone: 7397469288
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

## 2024_28_III_OOPS Using Java Lab

### REC_2028_OOPS using Java_Week 7_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

### Section 1 : Coding

1. Problem Statement

Alex and Bob are designing a control system for household appliances, and one of the appliances is a washing machine. You want to create a program to help them that models the washing machine as a motor and calculates its electricity consumption based on its capacity.

Define an interface named Motor with the following methods:

void run() double consume(double capacity)

Create a class called WashingMachine that implements the Motor interface.

In the WashingMachine class:

Implement the run() method to print "Washing machine is running."Implement a consume() method to print "Washing machine is consuming electricity."Implement the consume(double capacity) method to calculate the electricity consumption (in kWh) of the washing machine based on its capacity. The formula for electricity consumption is (capacity * 0.05).

*Input Format*

The input consists of a double value representing the capacity of the washing machine in kW.

*Output Format*

The first line of output prints "Washing machine is running."

The second line prints "Washing machine is consuming electricity."

The third line prints "Electricity consumption: X kWh" where X is a double value, rounded off to two decimal places, representing the electricity consumption.

Refer to the sample output for formatting specifications.

*Sample Test Case*

Input: 2.5

Output: Washing machine is running.
Washing machine is consuming electricity.
Electricity consumption: 0.13 kWh

*Answer*

```
import java.util.Scanner;

interface Motor {
   void run();
   double consume(double capacity);
}
class WashingMachine implements Motor {
   public void run() {
      System.out.println("Washing machine is running.");
   }
   public void consume() {
      System.out.println("Washing machine is consuming electricity.");
```

```
  }
  public double consume(double capacity) {
    return capacity * 0.05;
  }
  }

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

    WashingMachine washingMachine = new WashingMachine();

    double capacity = scanner.nextDouble();

    washingMachine.run();
    washingMachine.consume();


    double consumption = washingMachine.consume(capacity);
    System.out.printf("Electricity consumption: %.2f kWh" ,consumption);

    scanner.close();
  }
}
```

***Status :*** Correct ***Marks : 10/10***

2. Problem Statement:

Sam is developing a geometry application and needs a class for trapezoid calculations. Create a "Trapezoid" class implementing a "ShapeInput" interface with a method to input trapezoid dimensions.

Also, implement a "ShapeCalculator" interface with methods to compute area and perimeter. In the "Main" class, instantiate Trapezoid, gather user input, and display the calculated area and perimeter with two decimal places.

Note

Area of Trapezoid = (1/2) * (base1 + base2) * height

Perimeter of Trapezoid = base1 + base2 + side1 + side2

*Input Format*

The first line of input is a double-point value representing base1 of the trapezoid.

The second line of input is a double-point value representing base2 of the trapezoid.

The third line of input is a double-point value representing the height of the trapezoid.

The fourth line of input is a double-point value representing side1 of the trapezoid.

The fifth line of input is a double-point value representing side2 of the trapezoid.

*Output Format*

The output displays the two lines of the calculated area (double type) and perimeter (double type) of the trapezoid, each rounded to two decimal places in the following format:

"Area of the Trapezoid: <<calculated area>>".

Perimeter of the Trapezoid: <<calculated perimeter>>".

Refer to the sample output for the formatting specifications.

*Sample Test Case*

Input: 1.0
2.0
1.0
3.0
1.0
Output: Area of the Trapezoid: 1.50
Perimeter of the Trapezoid: 7.00

*Answer*

```
import java.util.Scanner;

interface ShapeInput {
    void getInput();
    }
interface ShapeCalculator {
    double calculateArea();
    double calculatePerimeter();
}
class Trapezoid implements ShapeInput, ShapeCalculator {
    private double base1, base2, height, side1, side2;
    public void getInput() {
        Scanner sc = new Scanner(System.in);
        base1 = sc.nextDouble();
        base2 = sc.nextDouble();
        height = sc.nextDouble();
        side1 = sc.nextDouble();
        side2 = sc.nextDouble();
    }
    public double calculateArea() {
        return 0.5 * (base1 + base2) * height;
    }
    public double calculatePerimeter() {
        return base1 + base2 + side1 + side2;
    }
}

public class Main {
    public static void main(String[] args) {
        Trapezoid trapezoid = new Trapezoid();
        trapezoid.getInput();

        double area = trapezoid.calculateArea();
        double perimeter = trapezoid.calculatePerimeter();

        System.out.println("Area of the Trapezoid: " + String.format("%.2f", area));
        System.out.println("Perimeter of the Trapezoid: " + String.format("%.2f",
perimeter));
    }
}
```

***Status :*** Correct ***Marks : 10/10***

## 3. Problem Statement

Jeevan is developing a fitness-tracking application to monitor daily physical activity.

The application incorporates a FitnessTracker class that implements two interfaces: StepCounter for tracking the number of steps taken and CalorieCalculator for estimating total calories burned based on total steps.

Jeevan needs your help creating a program.

Note

The calorie calculation formula is: Total caloriesBurned = (total steps / 100.0) * 20.0.

***Input Format***

The first line of input is an integer n, representing the number of days Jeevan wants to input data.

The second line consists of space-separated integers, representing the number of steps Jeevan took on each day.

***Output Format***

The first line of output prints: "Total Steps: <totalSteps>", where '<totalSteps>' is the sum of steps (integer) taken over 'n' days.

The second line prints: "Calories Burned: <caloriesBurned>", where '<caloriesBurned>' is the estimated total calories (double-point number) burned based on the total steps taken rounded off to two decimal places.

Refer to the sample output for the formatting specifications.

***Sample Test Case***

Input: 3
340 234 987

Output: Total Steps: 1561

```
Calories Burned: 312.20
```

***Answer***

```
import java.util.Scanner;

interface StepCounter {
    void countSteps(int steps);
    int getTotalSteps();
    }
interface CalorieCalculator {
    double calculateCaloriesBurned(int totalSteps);
}
class FitnessTracker implements StepCounter, CalorieCalculator {
    private int totalSteps = 0;
    public void countSteps(int steps) {
        totalSteps += steps;
    }
    public int getTotalSteps() {
        return totalSteps;
    }
    public double calculateCaloriesBurned(int totalSteps) {
        return (totalSteps / 100.0) * 20.0;
    }
}

class Main
{

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

        FitnessTracker tracker = new FitnessTracker();

        int n = scanner.nextInt();

        for (int i = 0; i < n; i++) {
            int steps = scanner.nextInt();
            tracker.countSteps(steps);
        }

        int totalSteps = tracker.getTotalSteps();
        System.out.println("Total Steps: " + totalSteps);
        double caloriesBurned = tracker.calculateCaloriesBurned(totalSteps);
        System.out.printf("Calories Burned: %.2f%n", caloriesBurned);

        scanner.close();
    }
}
```

***Status :*** Correct ***Marks : 10/10***

4. Problem Statement:

Ray is developing a tax calculation program in Java. The program includes an interface named TaxCalculator with a method to calculate tax based on salary. The SimpleTaxCalculator class implements this interface and determines the tax to be paid based on the salary amount using progressive tax slabs.

Your task is to implement this system. The program first takes an integer T representing the number of test cases, followed by T salary values. For each salary, calculate the total tax to be paid based on the following progressive tax rules:

For the first  50,000 of salary, the tax rate is 5%.For the next  50,000 (i.e., from  50,001 to  1,00,000), the tax rate is 10%.For any amount above  1,00,000, the tax rate is 20%. (That is, only the amount above  1,00,000 is taxed at 20%.)

Example

Input

3

78000

110000

23000

Output

5300

9500

1150

Explanation

For Salary Rs. 78,000

Tax = 0.1 * (78,000 - 50,000) + 0.05 * 50,000 = 5,300

For Salary Rs. 1,10,000

Tax = 0.2 * (110000 - 100000)+ 0.1 * 50,000 + 0.05 * 50,000 = 9,500

For Salary Rs. 23,000

Tax = 0.05 * 23,000 = 1,150

*Input Format*

The first line of the input consists of an integer, T, representing the number of test cases.

The next T lines of the input consist of a single integer, representing the annual salary of an individual, separated by a line.

*Output Format*

The output displays the calculated tax as an integer for each test case, separated by a line.

Refer to the sample output for the formatting specifications.

*Sample Test Case*

Input: 2
100
300

Output: 5
15

*Answer*

```
import java.util.Scanner;
```

```
interface TaxCalculator{
    int calculateTax(double salary);
    }
class SimpleTaxCalculator implements TaxCalculator {
    public int calculateTax(double salary) {
        double tax = 0;
        if (salary <= 50000) {
            tax = salary * 0.05;
        }
        else if (salary <= 100000) {
            tax = 50000 * 0.05 + (salary - 50000) * 0.10;
        }
        else {
            tax = 50000 * 0.05 + 50000 * 0.10 + (salary - 100000) * 0.20;
        }
        return (int) tax;
}
}

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

        TaxCalculator taxCalculator = new SimpleTaxCalculator();

        for (int i = 0; i < T; i++) {
            int salary = scanner.nextInt();
            int tax = taxCalculator.calculateTax(salary);
            System.out.println(tax);
        }

        scanner.close();
    }
}
```

***Status :*** Correct ***Marks : 10/10***