

# Rajalakshmi Engineering College

Name: Zaina Raheen A.P

Email: 241801328@rajalakshmi.edu.in

Roll no: 241801328

Phone: 9384899474

Branch: REC

Department: AI & DS - Section 4

Batch: 2028

Degree: B.E - AI & DS

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 7\_Q1

Attempt : 1

Total Mark : 10

Marks Obtained : 10

### Section 1 : Coding

#### 1. Problem Statement:

Rajiv is analyzing the energy consumption in his household and wants to calculate the total cost based on the daily energy usage. He is given the rate per unit of electricity and the energy consumed for multiple days. To structure this calculation efficiently, he decides to use an interface-based approach.

Implement an interface CostCalculator with the necessary methods to retrieve energy details and compute the cost. The calculations should be handled in the EnergyConsumptionTracker class, while the EnergyConsumptionApp class should only handle input and output.

Formula

Energy Cost for one day = Energy Consumed per day \* Rate Per Unit

### ***Input Format***

The first line of input consists of the rate per unit as an 'R' (a double value).

The second line of input consists of the number of days 'N' (an integer).

The third line of input consists of the daily energy consumption values for each day 'D' (double values), separated by space.

### ***Output Format***

The first line of the output prints: "Day-wise Energy Cost:"

The next N lines of the output print the day-wise energy costs(double type) and the total energy cost (double type) in Indian Rupees in the following format: "Day [day\_number]: Rs. [energy\_cost]"

The last line of the output prints: "Total Energy Cost: Rs. [total\_cost]"

Note: energy\_cost and total\_cost are rounded off to two decimal points

Refer to the sample output for the formatting specifications.

### ***Sample Test Case***

Input: 0.01

3

10.0 20.0 30.0

Output: Day-wise Energy Cost:

Day 1: Rs. 0.10

Day 2: Rs. 0.20

Day 3: Rs. 0.30

Total Energy Cost: Rs. 0.60

### ***Answer***

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

```
interface CostCalculator {
```

```

    void getEnergyDetails(Scanner scanner);
    void calculateAndDisplayCost();
}

class EnergyConsumptionTracker implements CostCalculator {
    private double ratePerUnit;
    private int numDays;
    private double[] dailyConsumption;

    public EnergyConsumptionTracker(double ratePerUnit, int numDays) {
        this.ratePerUnit = ratePerUnit;
        this.numDays = numDays;
        this.dailyConsumption = new double[numDays];
    }

    @Override
    public void getEnergyDetails(Scanner scanner) {
        for (int i = 0; i < numDays; i++) {
            dailyConsumption[i] = scanner.nextDouble();
        }
    }

    @Override
    public void calculateAndDisplayCost() {
        System.out.println("Day-wise Energy Cost:");
        double totalCost = 0.0;
        for (int i = 0; i < numDays; i++) {
            double cost = dailyConsumption[i] * ratePerUnit;
            totalCost += cost;
            System.out.printf("Day %d: Rs. %.2f%n", i + 1, cost);
        }
        System.out.printf("Total Energy Cost: Rs. %.2f%n", totalCost);
    }
}

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

        double ratePerUnit = scanner.nextDouble();
        int numDays = scanner.nextInt();

        CostCalculator tracker = new EnergyConsumptionTracker(ratePerUnit,
numDays);
    }
}

```

```
        tracker.getEnergyDetails(scanner);  
        tracker.calculateAndDisplayCost();  
    scanner.close();  
    }  
}
```

**Status :** Correct

**Marks :** 10/10

# Rajalakshmi Engineering College

Name: Zaina Raheen A.P  
Email: 241801328@rajalakshmi.edu.in  
Roll no: 241801328  
Phone: 9384899474  
Branch: REC  
Department: AI & DS - Section 4  
Batch: 2028  
Degree: B.E - AI & DS

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 8\_Q4

Attempt : 1  
Total Mark : 10  
Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

A local municipality is implementing an online voting system for a community event and wants to ensure that only eligible voters (those aged 18 or older) can participate.

Your task is to develop a program that validates the age of individuals attempting to vote online. If the user's age is below 18, the program should throw a custom exception, `InvalidAgeException`, preventing them from casting their vote. If the input is invalid, catch the appropriate `InputMismatchException` and print the in-built exception message.

##### *Input Format*

The input consists of an integer representing the age.

##### *Output Format*

If the age is 18 or older, print "Eligible to vote"

If the age is below 18, print "Exception occurred: InvalidAgeException: Age is not valid to vote"

If there is any other type of exception, print "An error occurred: " followed by the in-built exception message.

Refer to the sample output for formatting specifications.

### **Sample Test Case**

Input: 20

Output: Eligible to vote

### **Answer**

// You are using Java

import java.util.Scanner;

import java.util.InputMismatchException;

```
class InvalidAgeException extends Exception {  
    public InvalidAgeException(String message) {  
        super(message);  
    }  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        try {  
            int age = sc.nextInt();  
            validateAge(age);  
            System.out.println("Eligible to vote");  
        } catch (InvalidAgeException e) {  
            System.out.println("Exception occurred: InvalidAgeException: " +  
e.getMessage());  
        } catch (InputMismatchException e) {  
            System.out.println("An error occurred: " + e);  
        }  
    }  
}
```

```
public static void validateAge(int age) throws InvalidAgeException {  
    if (age < 18) {  
        throw new InvalidAgeException("Age is not valid to vote");  
    }  
}  
}
```

**Status :** Correct

**Marks :** 10/10