

Rajalakshmi Engineering College

Name: Brindha B
Email: 241501035@rajalakshmi.edu.in
Roll no: 241501035
Phone: 6369668750
Branch: REC
Department: AI & ML - Section 3
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results



2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 5_Q3

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Neha is working as a developer for CityElectricity Board, which wants to build a household electricity billing system.

Each customer's electricity account has:

A Customer ID (integer) A Customer Name (string) Units Consumed (double)

The electricity bill is calculated based on these rules:

For the first 100 units 5 units charge per unit
For the next 100 units (101–200) 7 units charge per unit
For units above 200 10 units charge per unit
If the total bill exceeds 2000 units, a 5% discount is applied on the final bill.

Neha has been asked to implement this system using:

A class with attributes for customer details.A constructor to initialize customer details.Setter methods to update details if needed.Getter methods to retrieve details.Objects of the class to represent customers.

Finally, display each customer's details and final bill amount.

Input Format

The first line of input contains an integer N, representing the number of customers.

For each customer:

- The next line contains the Customer ID (integer).
- The following line contains the Customer Name (string).
- The next line contains the Units Consumed (double).

Output Format

For each customer, print the details in the following format:

Customer ID: <customer_id>

Customer Name: <customer_name>

Final Bill: <final_bill> (rounded to one decimal place)

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1

1001

Ravi Kumar

80

Output: Customer ID: 1001

Customer Name: Ravi Kumar

Final Bill: 400.0

Answer

```
// You are using Java
```

```
import java.util.Scanner;

// Class to represent customer's electricity account
class ElectricityAccount {
    private int customerId;
    private String customerName;
    private double unitsConsumed;
    private double finalBill;

    // Constructor
    public ElectricityAccount(int customerId, String customerName, double unitsConsumed) {
        this.customerId = customerId;
        this.customerName = customerName;
        this.unitsConsumed = unitsConsumed;
        this.finalBill = calculateBill();
    }

    // Method to calculate bill based on consumption slabs
    private double calculateBill() {
        double bill = 0.0;
        double units = unitsConsumed;

        if (units <= 100) {
            bill = units * 5;
        } else if (units <= 200) {
            bill = (100 * 5) + ((units - 100) * 7);
        } else {
            bill = (100 * 5) + (100 * 7) + ((units - 200) * 10);
        }

        // Apply 5% discount if bill > 2000
        if (bill > 2000) {
            bill = bill - (bill * 0.05);
        }

        return bill;
    }

    // Display customer details and bill
    public void displayDetails() {
        System.out.println("Customer ID: " + customerId);
    }
}
```

```
        System.out.println("Customer Name: " + customerName);
        System.out.printf("Final Bill: %.1f\n", finalBill);
    }
}

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

        int N = Integer.parseInt(scanner.nextLine());
        ElectricityAccount[] customers = new ElectricityAccount[N];

        for (int i = 0; i < N; i++) {
            int customerId = Integer.parseInt(scanner.nextLine());
            String customerName = scanner.nextLine();
            double unitsConsumed = Double.parseDouble(scanner.nextLine());

            ElectricityAccount account = new ElectricityAccount(customerId,
customerName, unitsConsumed);
            customers[i] = account;
        }

        for (ElectricityAccount customer : customers) {
            customer.displayDetails();
        }

        scanner.close();
    }
}
```

Status : Correct

Marks : 10/10