Rajalakshmi Engineering College

Name: Darshan S
Email: 241801040@rajalakshmi.edu.in
Roll no: 241801040
Phone: 7305911089
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 5_Q2

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

You are working as a developer for CityBank, which wants to build a basic account management system.

Each customer at the bank has:

An Account Number (integer)A Customer Name (string)An Initial Balance (double)

The bank allows two types of transactions:

Deposit – increases the balance.Withdrawal – decreases the balance only if enough funds are available.

If the withdrawal amount is greater than the balance, the withdrawal should not happen, and the balance should remain the same.

You are required to implement this system using:

A class with attributes for account details.A constructor to initialize account 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 account details after all transactions.

Input Format

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

For each customer:

- The next line contains the account number (integer).
- The following line contains the customer name (string).
- The next line contains the initial balance (double).
- The next line contains the deposit amount (double).
- The next line contains the withdrawal amount (double).

Output Format

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

1. Account Number: <account_number>
2. Customer Name: <customer_name>
3. Final Balance: <final_balance> (rounded to one decimal place)

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
1234
Rahul Sharma
5000
2000
3000

Output: Account Number: 1234
Customer Name: Rahul Sharma

Final Balance: 4000.0

Answer

```
import java.util.Scanner;

public class Main {
   static class Customer {
      private int accountNumber;
      private String name;
      private double balance;

      public Customer(int accountNumber, String name, double initialBalance) {
         this.accountNumber = accountNumber;
         this.name = name;
         this.balance = initialBalance;
      }

      public int getAccountNumber() {
         return accountNumber;
      }

      public String getName() {
         return name;
      }

      public double getBalance() {
         return balance;
      }

      public void setName(String newName) {
         this.name = newName;
      }

      public void deposit(double amount) {
         if (amount > 0) {
            this.balance += amount;
         }
      }

      public void withdraw(double amount) {
         if (amount > 0 && amount <= this.balance) {
            this.balance -= amount;
            }
        }
    }

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

        try {
            int N = scanner.nextInt();
            scanner.nextLine();

            for (int i = 0; i < N; i++) {
                int accountNumber = scanner.nextInt();
                scanner.nextLine();
                String name = scanner.nextLine();
                double initialBalance = scanner.nextDouble();
                double depositAmount = scanner.nextDouble();
                double withdrawalAmount = scanner.nextDouble();
                // Removed the extra scanner.nextLine() here to prevent the error

                Customer customer = new Customer(accountNumber, name, 
initialBalance);

                customer.deposit(depositAmount);
                customer.withdraw(withdrawalAmount);

                System.out.println("Account Number: " + 
customer.getAccountNumber());
                System.out.println("Customer Name: " + customer.getName());
                System.out.printf("Final Balance: %.1f%n", customer.getBalance());
            }

        } catch (Exception e) {
            System.err.println("An error occurred with the input: " + e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

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

Rajalakshmi Engineering College

Name: Darshan S
Email: 241801040@rajalakshmi.edu.in
Roll no: 241801040
Phone: 7305911089
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 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 unitFor the next 100 units (101–200)   7 units charge per unitFor units above 200   10 units charge per unitIf 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;
import java.text.DecimalFormat;

public class Main {
    static class Customer {
        private int customerId;
        private String name;
        private double unitsConsumed;

        public Customer(int customerId, String name, double unitsConsumed) {
            this.customerId = customerId;
            this.name = name;
            this.unitsConsumed = unitsConsumed;
        }

        public int getCustomerId() {
            return customerId;
        }

        public String getName() {
            return name;
        }

        public double getUnitsConsumed() {
            return unitsConsumed;
        }

        public void setCustomerId(int customerId) {
            this.customerId = customerId;
        }

        public void setName(String name) {
            this.name = name;
        }

        public void setUnitsConsumed(double unitsConsumed) {
            this.unitsConsumed = unitsConsumed;
        }

        public double calculateBill() {
            double bill = 0;
            double remainingUnits = this.unitsConsumed;
            if (remainingUnits > 200) {
                bill += (remainingUnits - 200) * 10;
                remainingUnits = 200;
            }
            if (remainingUnits > 100) {
                bill += (remainingUnits - 100) * 7;
                remainingUnits = 100;
            }
            if (remainingUnits > 0) {
                bill += remainingUnits * 5;
            }

            if (bill > 2000) {
                bill *= 0.95;
            }

            return bill;
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        DecimalFormat df = new DecimalFormat("0.0");

        try {
            int N = scanner.nextInt();
            scanner.nextLine();

            for (int i = 0; i < N; i++) {
                int customerId = scanner.nextInt();
                scanner.nextLine();
                String name = scanner.nextLine();
                double unitsConsumed = scanner.nextDouble();
                if (i < N - 1) {
                    scanner.nextLine();
                }

                Customer customer = new Customer(customerId, name,
unitsConsumed);
                double finalBill = customer.calculateBill();
                System.out.println("Customer ID: " + customer.getCustomerId());
                System.out.println("Customer Name: " + customer.getName());
                System.out.println("Final Bill: " + df.format(finalBill));
            }
        } catch (Exception e) {
            System.err.println("An error occurred with the input: " + e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

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