

Rajalakshmi Engineering College

Name: shyam ganesh
Email: 241801266@rajalakshmi.edu.in
Roll no:
Phone: null
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

// You are using Java

import java.util.*;

```
class BankAccount {
    int accountNumber;
    String customerName;
    double balance;

    // Constructor
    BankAccount(int accountNumber, String customerName, double balance) {
        this.accountNumber = accountNumber;
        this.customerName = customerName;
        this.balance = balance;
    }

    // Deposit money
    void deposit(double amount) {
        balance += amount;
    }

    // Withdraw money (only if enough balance)
    void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
        }
    }

    // Show account details
    void showDetails() {
        System.out.println("Account Number: " + accountNumber);
        System.out.println("Customer Name: " + customerName);
        System.out.printf("Final Balance: %.1f\n", balance);
    }
}

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

```

int N = sc.nextInt(); // number of customers
sc.nextLine();        // consume newline

for (int i = 0; i < N; i++) {
    int accNo = sc.nextInt();
    sc.nextLine(); // consume newline
    String name = sc.nextLine();
    double initBalance = sc.nextDouble();
    double depositAmt = sc.nextDouble();
    double withdrawAmt = sc.nextDouble();

    // create account
    BankAccount customer = new BankAccount(accNo, name, initBalance);

    // perform transactions
    customer.deposit(depositAmt);
    customer.withdraw(withdrawAmt);

    // show result
    customer.showDetails();
}

sc.close();
}
}

```

Status : Correct

Marks : 10/10