

Rajalakshmi Engineering College

Name: KRITHESHWARAN R
Email: 241901049@rajalakshmi.edu.in
Roll no: 241901049
Phone: 9843565002
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

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.*;

class Customer{
    private int accno;
    private String name;
    private double initial;
    private double balance;

    public Customer(int accno,String name,double initial){//construcor
        this.accno = accno;
        this.name = name;
        this.initial = initial;
    }
    public int getAccnum(){//to access or get values
        return accno;
    }
    public String getCusname(){
        return name;
    }
    //setAccnum and setCusname are optional
    public double getBalance(){
        return balance;
    }

    public void setbalance(double balance,double withdrawal){//set to update
        balance value
        if(balance>=withdrawal){
            this.balance =balance -withdrawal;
        }
        else
            this.balance=balance;
    }
}

public class Main{
    public static void main(String[] args){
        Scanner sc=new Scanner(System.in);
        int N=sc.nextInt();
        Customer []cus = new Customer[N];
    }
}
```

```
//array creation "Customer" userdefined dtype created by class
for(int i=0;i<N;i++){
```

```
    int accno=sc.nextInt();
    sc.nextLine();
    String name=sc.nextLine();
    double initial=sc.nextDouble();
    double deposit=sc.nextDouble();
    double withdrawal=sc.nextDouble();
    double balance = initial + deposit;
```

```
    cus[i] = new Customer(accno, name, initial);
```

```
//cus[i] is already of type Customer because the array was declared
```

```
//When you have an array of objects, like Customer[] cus, you don't repeat the
type every time.
```

```
//so only we here ignoring the syntax classname here.
```

```
    cus[i].setbalance(balance,withdrawal);
    System.out.println("Account Number: "+cus[i].getAccnum());
    System.out.println("Customer Name: "+cus[i].getCusname());
    System.out.println("Final Balance: "+cus[i].getBalance());
```

```
    }
}
}
```

Status : Correct

Marks : 10/10