

# Rajalakshmi Engineering College

Name: KISHORE RAJ.A.S

Email: 241801127@rajalakshmi.edu.in

Roll no: 241801127

Phone: 7397295789

Branch: REC

Department: AI & DS - Section 3

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 {
    private int accountNumber;
    private String customerName;
    private double balance;

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

    public void setAccountNumber(int accountNumber) {
        this.accountNumber = accountNumber;
    }

    public void setCustomerName(String customerName) {
        this.customerName = customerName;
    }

    public void setBalance(double balance) {
        this.balance = balance;
    }

    public int getAccountNumber() {
        return accountNumber;
    }

    public String getCustomerName() {
        return customerName;
    }

    public double getBalance() {
        return balance;
    }
}
```

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

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

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

        if (!sc.hasNextInt()) {
            return;
        }
        int n = sc.nextInt();
        sc.nextLine();

        List<BankAccount> customers = new ArrayList<>();

        for (int i = 0; i < n; i++) {
            if (!sc.hasNextInt()) {
                break;
            }
            int accNo = sc.nextInt();
            sc.nextLine();

            if (!sc.hasNextLine()) {
                break;
            }
            String name = sc.nextLine();

            if (!sc.hasNextDouble()) {
                break;
            }
            double initialBalance = sc.nextDouble();
        }
    }
}
```

```
if (!sc.hasNextDouble()) {
    break;
}
double depositAmount = sc.nextDouble();

if (!sc.hasNextDouble()) {
    break;
}
double withdrawAmount = sc.nextDouble();
BankAccount account = new BankAccount(accNo, name, initialBalance);
account.deposit(depositAmount);
account.withdraw(withdrawAmount);
customers.add(account);
}

for (BankAccount account : customers) {
    System.out.printf("Account Number: %d\n",
account.getAccountNumber());
    System.out.printf("Customer Name: %s\n", account.getCustomerName());
    System.out.printf("Final Balance: %.1f\n", account.getBalance());
}
}
```

**Status :** Correct

**Marks :** 10/10