

19/01/24

## WEEK-5

Date	20
Page No.	10
Page No.	A4

Q) Develop a java program to create a class Bank that maintains two kinds of accounts for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if balance falls below this level, a service charge is imposed. Create a class account that stores customer name, account number and type of account. From this derive the classes cur-acct and sav-acct to make them more specific to their requirement. Include the necessary methods in order to achieve following:

- a) accept deposit from customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest.
- d) permit withdrawal and update balance. Check for minimum balance, impose penalty if necessary and update balance.

WANT TO MAKE IT EASY

NOT HARD BUT IT PRACTICE WITH SMART

IDEA TO MAINTAIN AND READ

QUESTION, PRACTICE

CODE

```
import java.util.Scanner;  
class Account {  
    String customerName;  
    long accno;  
    String accountType;  
    double balance;  
    public Account(String customerName, long accno, String  
        account type) {  
        this.customerName = customerName;  
        this.accno = accno;  
        this.accountType = account type;  
        this.balance = 0.0;  
    }  
    public void displayBalance() {  
        System.out.println("Account Number: " + accno);  
        System.out.println("Customer Name: " + customerName);  
        System.out.println("Account Type: " + accountType);  
        System.out.println("Balance: $" + balance);  
    }  
}
```

```
class Current extends Account {  
    double minbalance;  
    double service_charge;  
    public Current(String customerName, long accno) {  
        super(customerName, accno, "Current");  
        this.minbalance = 500.0;  
        this.service_charge = 50.0;  
    }  
}
```

```
public void withdraw (double amount) {  
    if (balance - amount >= minBalance) {  
        balance -= amount;  
        System.out.println ("Withdrawal successful. Current  
        Balance : $" + balance);  
    } else {  
        System.out.println ("Insufficient funds.  
        Withdrawal not allowed");  
    }  
}
```

```
public void  
imposeServiceCharge() {  
    if (balance < minBalance) {  
        balance -= serviceCharge;  
        System.out.println ("Service charge imposed.  
        Current balance is Ro." + balance);  
    }  
}
```

```
class SavAcct extends Account {  
    double interestRate;  
    public SavAcct (String customerName, long accountNo) {  
        super (customerName, accountNo, "Savings");  
        this.interestRate = 0.05; // initial interest rate  
    }  
}
```

```
public void depositInterest() {  
    double interest = balance * interestRate;  
    balance += interest;  
}
```

M	T	W	T	F	S
8-2					
Project					A-4

System.out.println("Interest Deposited. Current Balance : \$ " + balance);

}

```
public void CompoundInterest(double initialAmount, int term) {
    double compoundInterest = initialAmount * Math.pow((1 + interestRate), term) - initialAmount;
    balance += compoundInterest;
    System.out.println("Compound Interest Deposited. Current Balance : Rs. " + balance);
}
```

public class Bank {

```
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.println("Choose account type : ");
    System.out.println("1 - Current");
    System.out.println("2 - Savings");
    System.out.println("Enter choice (1 or 2) : ");
    int choice = scanner.nextInt();
}
```

System.out.print("Enter customer name : ");

String customerName = scanner.next();

System.out.print("Enter account number : ");

long accNo = scanner.nextLong();

if (choice == 1) {

CurrAcc currAccount = new CurrAcc(customerName, accNo);

System.out.print("Enter initial balance : \$ ");

double initialBalance = scanner.nextDouble();

CurrAcc.balance = initialBalance;

System.out.print("Enter withdrawal : \$ ");

Topic 06

```
double withdrawalAmount = scanner.nextDouble();
CurrentAccount::withdraw(withdrawalAmount);
CurrentAccount::imposeServiceCharge();
CurrentAccount::displayBalance();
```

else if (choice == 2) {  
 SavingsAccount savaAcct("CustomerName", accno);  
 System.out.println("Enter initial balance : \$");  
 double initialBalance = scanner.nextDouble();  
 savaAcct.balance = initialBalance;  
 System.out.println("Enter withdrawal amount : \$");  
 double withdrawalAmount = scanner.nextDouble();  
 savaAcct.balance -= withdrawalAmount;  
 System.out.println("Withdrawal Successful.  
 Current balance : \$" + savaAcct.balance);  
 System.out.println("Enter interest rate");  
 double interestRate = scanner.nextDouble();  
 savaAcct.interestRate = interestRate;  
 savaAcct.display("balance");  
 System.out.println("Enter term in days for calculation");  
 int term = scanner.nextInt();  
 savaAcct.compoundInterest(initialBalance, term);  
 savaAcct.display("balance");

else {  
 System.out.println("Invalid choice");

## Output:

1. Savings Account

2. Current Account

2

Enter your name

Hansika

Enter balance amount

250000

Name: Hansika

Cheque Service Available!

Do you want to deposit (1.yes 2.No),

1

Enter amount to be deposited

2000

Current balance: 252000.0

Enter amount to be withdrawn

500

Withdrawn: 500

Current balance: 251500.0

SS  
19/1/24

ARUGUNTA HAMSIKA

1BM22CS054

Enter the number of users: 1

User 1

Enter customer name: athmica

Enter account number: 123

Enter initial deposit amount: INR 5000

Enter account type (Savings/Current): savings

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit

Enter your choice: 1

Enter account number: 123

Enter deposit amount: INR 200

Deposit of INR 200.0 successful

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit

Enter your choice: 2

Enter account number: 123

Enter withdrawal amount: INR 100

Withdrawal of INR 100.0 successful

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit

Enter your choice: 3  
Enter account number: 123  
Account Number: 123  
Customer Name: athmica  
Account Type: savings  
Balance: INR 5100.0

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit

Enter your choice: 4

Enter account number (for Savings account): 123  
Interest of INR 255.0 added to the account

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Compute Interest (Savings only)
5. Exit

Enter your choice: 5