NAME- Harshita Totala ROLL NO- SEITA14 DIV- IT A
SUBJECT- Object Oriented Programming

ASSIGNMENT NO. 9

TITLE: Case Study

AIM: Using concepts of Object Oriented programming develop solution for any one application-

1) Banking solution contains following operations such as 1. Create an account 2.Deposit money 3.Withdraw money 4. Honor daily withdrawal limit 5. Check the balance 6. Display Account information.

2) Inventory management contains following operations such as List of all products 2. Display individual product information 3. Purchase 4. Shipping 5.Balance stock 6. Loss and Profit calculation.

OBJECTIVE: To implement real time context.

SAMPLE CODE:

Create following classes and Methods

class Account : Set balance(),Getbalance(),setAccount type(),getaccounttype

Class Bank: creatAccount(),withdrawAmount(),depositAmount(),displayinfo()

Class customer: getCustomerName(),setCustomerName(),getcustomerAge(),setCustomerAge()

Class saving account: setMinimumBalance (),withdraw()

Input:

Enter your name: Sai

Enter your age: 15

Minimum age should be 18 to create an account.

Please enter valid age: 21

Enter your account Id: 1

Enter your account type: savings

Enter balance: 10000

Enter minimum balance: 1000

Output :

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 1

Enter your name: Sai

Enter your age: 15

Minimum age should be 18 to create an account.

Please enter valid age: 21

Enter your account Id: 1

Enter your account type: savings

Enter balance: 10000

Enter minimum balance: 1000

Do you want to perform more actions? (yes/no): yes

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 2

Welcome Sai Pande! Following are your account details:

Age :21

Account Id: 1

Account Type: savings

Balance: 10000.0

Minimum balance: 1000.0

Do you want to perform more actions? (yes/no): yes

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 3

Balance is: 10000.0

Do you want to perform more actions? (yes/no): yes

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 4

Enter the amount you want to deposit: 20000

Amount deposited successfully. Balance is: 30000.0

Do you want to perform more actions? (yes/no): yes

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 5

Enter the amount you want to withdraw: 30000

Withdrawal failed. Maximum limit of withdrawal in one transaction is Rs.20000.

Do you want to perform more actions? (yes/no): yes

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 5

Enter the amount you want to withdraw: 15000

PROGRAM-

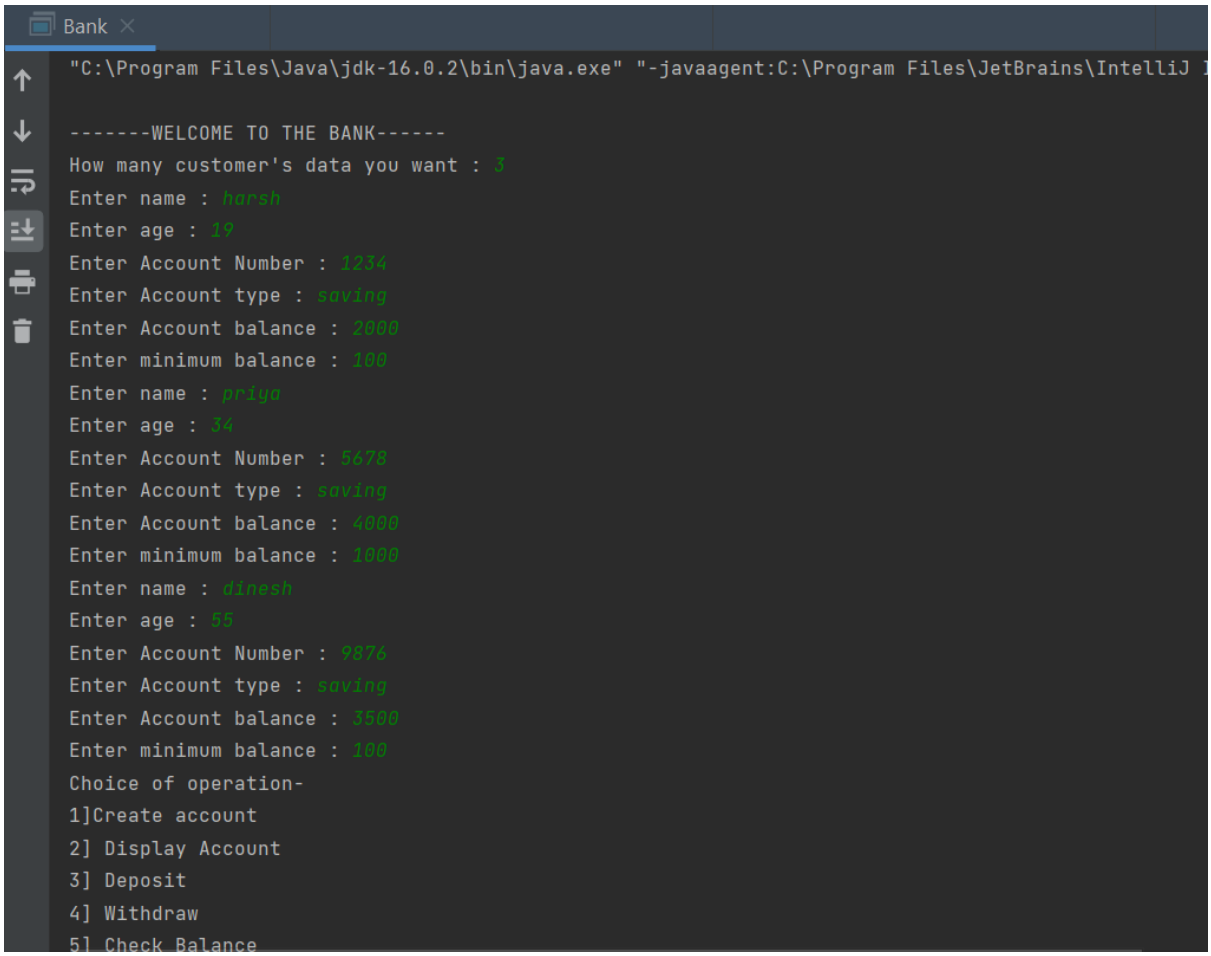
```
package com.company.assignment;
import java.util.Scanner;

class Bank_Acc{
    Scanner sc = new Scanner(System.in);
    String name;
    int age;
    String account_no;
    double balance;
    double min_balance;
    String acc_type;

    public void create_acc(){
        System.out.print("Enter name : ");
        name = sc.nextLine();
        System.out.print("Enter age : ");
        age = sc.nextInt();
        if (age<18){
            do{
                System.out.print("Minimum Age Should be 18.Enter you Age
again.");
                age = sc.nextInt();
            }while (age<18);
        }
        System.out.print("Enter Account Number : ");
        account_no = sc.next();
        System.out.print("Enter Account type : ");
        acc_type = sc.next();
        System.out.print("Enter Account balance : ");
        balance = sc.nextDouble();
        System.out.print("Enter minimum balance : ");
        min_balance = sc.nextDouble();
    }

    public void display_acc(){
        System.out.println("****************************************");
        System.out.println("WELCOME ! Details of Customer is-");
        System.out.println("NAME- "+name);
        System.out.println("ACCOUNT NUMBER- "+account_no);
        System.out.println("ACCOUNT TYPE- "+acc_type);
        System.out.println("BALANCE- "+balance);
        System.out.println("****************************************");

    }

    public void deposit(){
        long amt;
        System.out.print("Enter Amount to Deposit : ");
        amt = sc.nextLong();
        balance = balance + amt;
        System.out.println("AMOUNT DEPOSITED SUCCESSFULLY! NEW BALANCE-
```

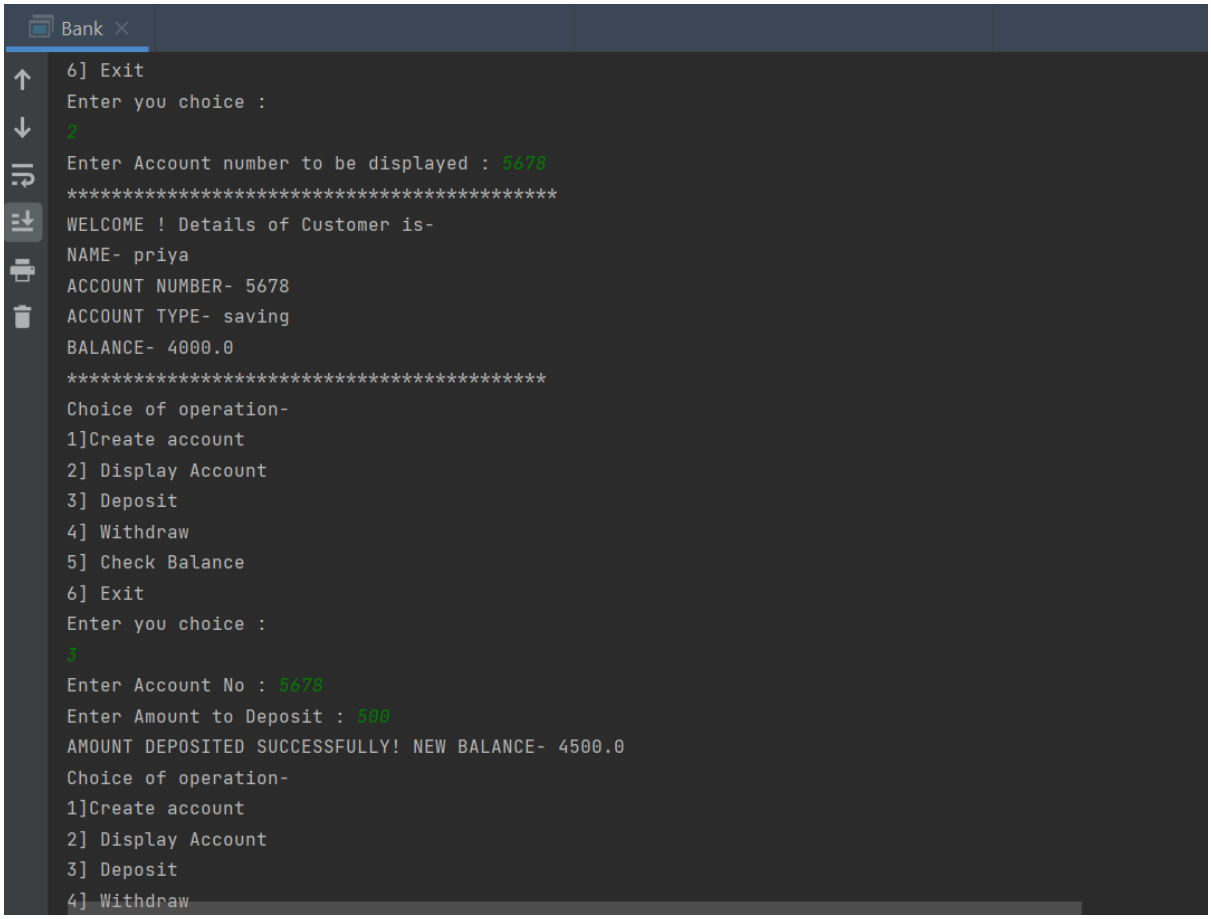
```
"+balance);
    }

    public void withdraw(){
        long amt;
        System.out.print("Enter Amount to Withdraw : ");
        amt = sc.nextLong();

        if((balance-amt)>min_balance)
        {
            if (balance >= amt) {
                balance = balance - amt;
                System.out.println("AMOUNT WITHDRAW SUCCESSFULLY! NEW
BALANCE- "+balance);
            }
        }
        else{
            System.out.println("MINIMUM BALANCE OF "+min_balance+"IS TO BE
MAINTAINED!");
            System.out.println("TRANSACTION FAILED! ");
        }
    }

    public void get_balance(){
        System.out.println("Current Balance is- "+balance);
    }

    boolean search(String acn) {
        if (account_no.equals(acn)) {
            return (true);
        }
        return (false);
    }

}
public class Bank {
    public static void main(String[] arg)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("\n-------WELCOME TO THE BANK------");
        int n;
        System.out.print("How many customer's data you want : ");
        n = sc.nextInt();
        Bank_Acc[] ac = new Bank_Acc[n];

        for (int i = 0; i < ac.length; i++) {
            ac[i] = new Bank_Acc();
            ac[i].create_acc();
        }
        int ch;
        boolean found=false;
        do {
            System.out.println("Choice of operation- ");
            System.out.println("1]Create account\n2] Display Account\n3]
Deposit\n4] Withdraw\n5] Check Balance\n6] Exit");
            System.out.println("Enter you choice :");
            ch = sc.nextInt();
            switch (ch) {

                case 1:
                    System.out.print("How many customer's data you want :
```

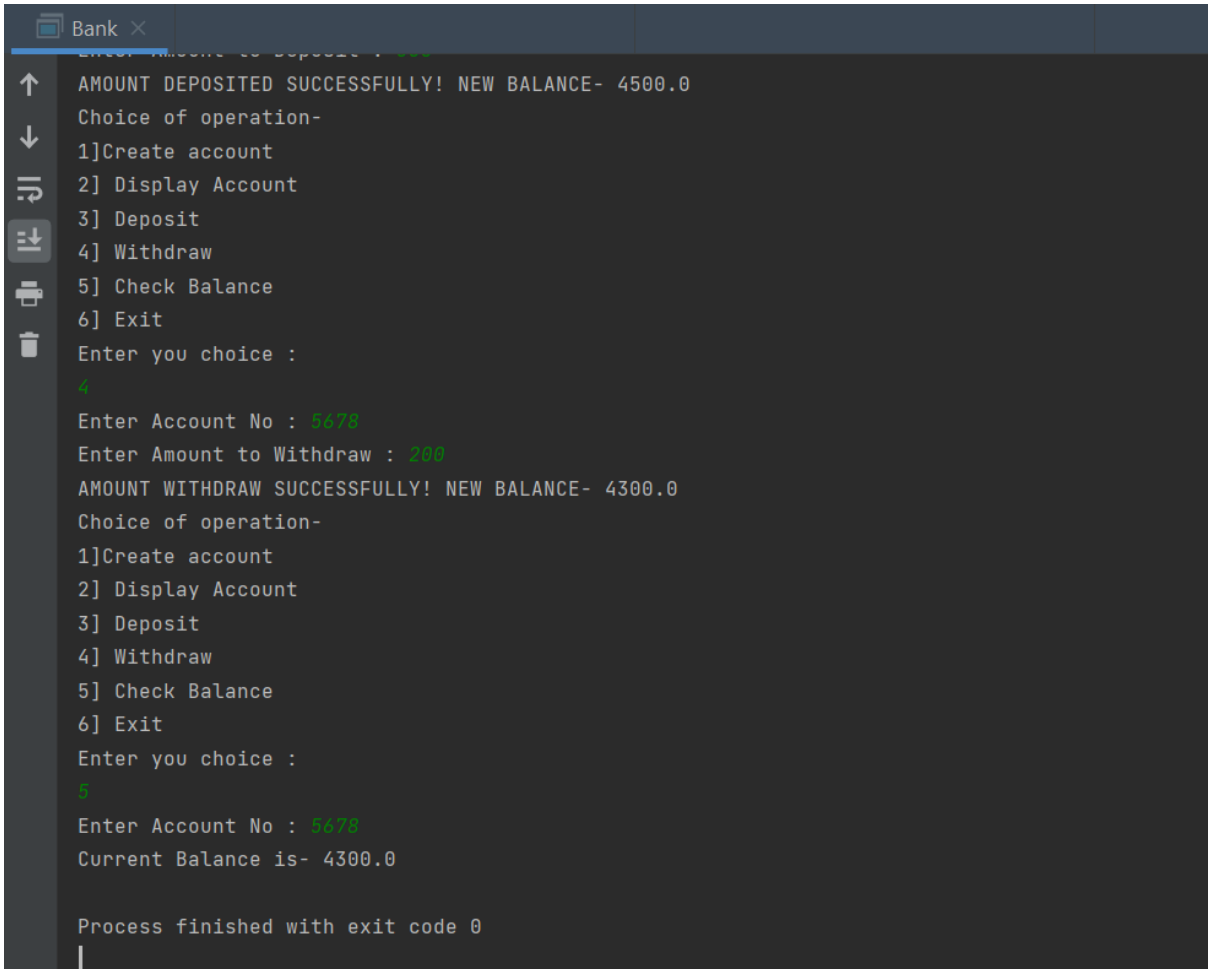
```
");
                    n = sc.nextInt();
                    ac = new Bank_Acc[n];

                    for (int i = 0; i < ac.length; i++) {
                        ac[i] = new Bank_Acc();
                        ac[i].create_acc();
                    }
                    break;


                    case 2:
                    System.out.print("Enter Account number to be displayed
: ");
                    String acn = sc.next();
                    for (int i = 0; i < ac.length; i++) {
                        found = ac[i].search(acn);
                        if (found) {
                            ac[i].display_acc();
                            break;
                        }
                    }
                    if (!found) {
                        System.out.println("Account Does Not Exist !!!");
                    }
                    break;


                case 3:
                    System.out.print("Enter Account No : ");
                    acn = sc.next();
                    found = false;
                    for (int i = 0; i < ac.length; i++) {
                        found = ac[i].search(acn);
                        if (found) {
                            ac[i].deposit();
                            break;
                        }
                    }
                    if (!found) {
                        System.out.println("Account Does Not Exist !!!");
                    }
                    break;

                case 4:
                    System.out.print("Enter Account No : ");
                    acn = sc.next();
                    found = false;
                    for (int i = 0; i < ac.length; i++) {
                        found = ac[i].search(acn);
                        if (found) {
                            ac[i].withdraw();
                            break;
                        }
                    }
                    if (!found) {
                        System.out.println("Account Does Not Exist !!!");
                    }
                    break;

                case 5:
```

```
                    System.out.print("Enter Account No : ");
                    acn = sc.next();
                    found = false;
                    for (int i = 0; i < ac.length; i++) {
                        found = ac[i].search(acn);
                        if (found) {
                            ac[i].get_balance();
                            break;
                        }
                    }
                    if (!found) {
                        System.out.println("Account Does Not Exist !!!");
                    }
                    break;


                case 6:
                    System.out.println("Thank You!! Do visit again.");
                    break;
            }
        }
        while (ch != 5);
    }
}
```

OUTPUT-


Bank
"C:\Program Files\Java\jdk-16.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ
-------WELCOME TO THE BANK------
How many customer's data you want : 3
Enter name : harsh
Enter age : 19
Enter Account Number : 1234
Enter Account type : saving
Enter Account balance : 2000
Enter minimum balance : 100
Enter name : priya
Enter age : 34
Enter Account Number : 5678
Enter Account type : saving
Enter Account balance : 4000
Enter minimum balance : 1000
Enter name : dinesh
Enter age : 55
Enter Account Number : 9876
Enter Account type : saving
Enter Account balance : 3500
Enter minimum balance : 100
Choice of operation-
1]Create account
2] Display Account
3] Deposit
4] Withdraw
5] Check Balance


Bank
6] Exit
Enter you choice :
2
Enter Account number to be displayed : 5678
****
WELCOME ! Details of Customer is-
NAME- priya
ACCOUNT NUMBER- 5678
ACCOUNT TYPE- saving
BALANCE- 4000.0
****
Choice of operation-
1]Create account
2] Display Account
3] Deposit
4] Withdraw
5] Check Balance
6] Exit
Enter you choice :
3
Enter Account No : 5678
Enter Amount to Deposit : 500
AMOUNT DEPOSITED SUCCESSFULLY! NEW BALANCE- 4500.0
Choice of operation-
1]Create account
2] Display Account
3] Deposit
4] Withdraw



CONCLUSION- Thus we performed the Bank Management with concepts of Object Oriented Programming.