**Student Name: Diptesh Ravindra Varule Roll No.:23277**

**Assignment No 9**

**Case Study 1**

**Problem Statement :**

**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.**

**Source Code:**

**Class Containing Main:**

**package** casestudy1;

**import** java.util.Scanner;

**public** **class** BankMain

{

**public** **static** Scanner *sc*=**new** Scanner(System.***in***);

**public** SavingsAccount a=**new** SavingsAccount();

**public** Customer c=**new** Customer();

**public** SavingsAccount createAccount()

{

*sc*.nextLine();

System.***out***.print("Enter your name: ");

String customername=*sc*.nextLine();

System.***out***.print("Enter your age: ");

**int** customerage=*sc*.nextInt();

**if**(customerage<18)

{

**do**{

System.***out***.print("Minimum age should be 18 to create an account.\nPlease enter valid age: ");

customerage=*sc*.nextInt();

}**while**(customerage<18);

}

System.***out***.print("Enter your account Id: ");

**int** accountid=*sc*.nextInt();

System.***out***.print("Enter your account type: ");

String accounttype=*sc*.next();

System.***out***.print("Enter balance: ");

**double** balance=*sc*.nextDouble();

System.***out***.print("Enter minimum balance: ");

**double** minbalance=*sc*.nextDouble();

c.setCustomerName(customername);

c.setCustomerAge(customerage);

a.setAccountId(accountid);

a.setAccountType(accounttype);

a.setBalance(balance);

a.setMinimumBalance(minbalance);

a.setCustomerObject(c);

**return** a;

}

**public** **void** getWithdrawAmount()

{

System.***out***.print("Enter the amount you want to withdraw: ");

**double** amount=*sc*.nextDouble();

**if**(amount>20000)

{

System.***out***.println("Withdrawal failed. Maximum limit of withdrawal in one transaction is Rs.20000.");

}

**else**

{

**if**(a.withdraw(amount)==**true**)

{ System.***out***.println("Withdrawal successful. Balance is: "+a.getBalance());

}

**else** System.***out***.println("Sorry!!! Not enough balance");

}

}

**public** **void** depositAmount(**double** amount){

**double** bal=a.getBalance()+amount;

a.setBalance(bal);

System.***out***.println("Amount deposited successfully. Balance is: "+a.getBalance());

}

**public** **void** checkBalance(){

System.***out***.println("Balance is: "+a.getBalance());

}

**public** **void** displayAccountInformation(){

System.***out***.println("Welcome "+c.getCustomerName()+"! Following are your account details:");

System.***out***.println("Age :"+c.getCustomerAge());

System.***out***.println("Account Id: "+a.getAccountId());

System.***out***.println("Account Type: "+a.getAccountType());

System.***out***.println("Balance: "+a.getBalance());

System.***out***.println("Minimum balance: "+a.getMinimumBalance());

}

**public** **static** **void** main(String[] args)

{ SavingsAccount a;

BankMain bm=**new** BankMain();

String ans;

**do**{

System.***out***.println("\n1.Create Account\n2.Display Account\n3.Check Balance\n4.Deposit Amount\n5.Withdraw Amount");

System.***out***.print("Enter your choice: ");

**int** choice=*sc*.nextInt();

System.***out***.println("");

**switch**(choice)

{

**case** 1:a=bm.createAccount();

**break**;

**case** 2:bm.displayAccountInformation();

**break**;

**case** 3:bm.checkBalance();

**break**;

**case** 4:System.***out***.print("Enter the amount you want to deposit: ");

**double** amount=*sc*.nextDouble();

bm.depositAmount(amount);

**break**;

**case** 5:bm.getWithdrawAmount();

**break**;

**default**:**break**;

}

System.***out***.print("\nDo you want to perform more actions? (yes/no): ");

ans=*sc*.next();

}**while**(ans.equalsIgnoreCase("yes"));

}

}

**Abstract class Account:**

**package** casestudy1;

**abstract** **public** **class** Account {

**protected** **double** balance;

**protected** **int** accountId;

**protected** String accountType;

**protected** Customer custobj;

**public** **void** setBalance(**double** balance)

{ **this**.balance=balance;

}

**public** **double** getBalance()

{ **return** balance;

}

**public** **void** setAccountId(**int** accountId)

{ **this**.accountId=accountId;

}

**public** **int** getAccountId()

{ **return** accountId;

}

**public** **void** setAccountType(String accountType){

**this**.accountType=accountType;

}

**public** String getAccountType()

{ **return** accountType;

}

**public** **void** setCustomerObject(Customer custobj)

{ **this**.custobj=custobj;

}

**public** Customer getCustomerObject()

{ **return** custobj;

}

**public** **abstract** **boolean** withdraw(**double** amount);

}

**Class Customer:**

**package** casestudy1;

**public** **class** Customer {

**private** String customerName;

**private** **int** customerAge;

**public** **void** setCustomerName(String customerName){

**this**.customerName=customerName;

}

**public** String getCustomerName(){

**return** customerName;

}

**public** **void** setCustomerAge(**int** customerAge){

**this**.customerAge=customerAge;

}

**public** **int** getCustomerAge(){

**return** customerAge;

}

}

**Class SavingsAccount:**

**package** casestudy1;

**public** **class** SavingsAccount **extends** Account{

**private** **double** minimumBalance;

**public** **void** setMinimumBalance(**double** minimumBalance){

**this**.minimumBalance=minimumBalance;

}

**public** **double** getMinimumBalance(){

**return** minimumBalance;

}

**public** **boolean** withdraw(**double** amount){

**if**((balance-amount)>minimumBalance){

balance-=amount;

**return** **true**;

}

**else**

**return** **false**;

}

}

**Output :**

1.Create Account

2.Display Account

3.Check Balance

4.Deposit Amount

5.Withdraw Amount

Enter your choice: 1

Enter your name: Diptesh Ravindra Varule

Enter your age: 19

Enter your account Id: 64

Enter your account type: Savings

Enter balance: 40000

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 Diptesh Ravindra Varule! Following are your account details:

Age :19

Account Id: 64

Account Type: Savings

Balance: 40000.0

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: 40000.0

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: 10000

Amount deposited successfully. Balance is: 50000.0

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: 20000

Withdrawal successful. 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: 25000

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: 2

Welcome Diptesh Ravindra Varule! Following are your account details:

Age :19

Account Id: 64

Account Type: Savings

Balance: 30000.0

Minimum balance: 1000.0

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