package com.project.atm;

import java.text.DecimalFormat;

import java.util.Scanner;

public class BankAccount

{

private double currentBalance = 10000;

private double savingBalance = 10000;

private int customerNumber;

private int pinNumber;

Scanner input = new Scanner(System.in);

DecimalFormat moneyFormat = new DecimalFormat("'$' ###,##0.00");

public int setCustomerNumber(int customerNumber) {

this.customerNumber = customerNumber;

return customerNumber;

}

public int getCustomerNumber() {

return customerNumber;

}

public int setPinNumber(int pinNumber) {

this.pinNumber = pinNumber;

return pinNumber;

}

public int getPinNumber() {

return pinNumber;

}

public double getCheckingBalance() {

return currentBalance;

}

public double getSavingBalance() {

return savingBalance;

}

public double calcCheckingWithdraw(double amount) {

currentBalance = (currentBalance - amount);

return currentBalance;

}

public double calcSavingWithdraw(double amount) {

savingBalance = (savingBalance - amount);

return savingBalance;

}

public double calcCheckingDeposit(double amount) {

currentBalance = (currentBalance + amount);

return currentBalance;

}

public double calcSavingDeposit(double amount) {

savingBalance = (savingBalance + amount);

return savingBalance;

}

public void getCheckingWithdrawInput() {

System.out.println("Checking Account Balance:" + moneyFormat.format(currentBalance));

System.out.println("Amount you want to withdraw from Checking Account:");

double amount = input.nextDouble();

if ((currentBalance - amount) >=0) {

calcCheckingWithdraw(amount);

System.out.println("New Checking Account Balance:" + moneyFormat.format(currentBalance));

}

else {

System.out.println("Balance cannot be Negative." +"\n");

}

}

public void getsavingWithdrawInput() {

System.out.println("Saving Account Balance:" + moneyFormat.format(savingBalance));

System.out.println("Amount you want to withdraw from Saving Account:");

double amount = input.nextDouble();

if ((savingBalance - amount) >=0) {

calcSavingWithdraw(amount);

System.out.println("New Checking Account Balance:" + moneyFormat.format(savingBalance));

}

else {

System.out.println("Balance cannot be Negative." +"\n");

}

}

public void getCheckingDepositInput() {

System.out.println("Checking Account Balance:" + moneyFormat.format(currentBalance));

System.out.println("Amount you want to Deposite from Checking Account:");

double amount = input.nextDouble();

if ((currentBalance + amount) >=0) {

calcCheckingDeposit(amount);

System.out.println("New Checking Account Balance:" + moneyFormat.format(currentBalance));

}

else {

System.out.println("Balance cannot be Negatrive." +"\n");

}

}

public void getSavingDepositInput() {

System.out.println("Saving Account Balance:" + moneyFormat.format(savingBalance));

System.out.println("Amount you want to Deposite from Saving Account:");

double amount = input.nextDouble();

if ((savingBalance + amount) >=0) {

calcSavingDeposit(amount);

System.out.println("New Saving Account Balance:" + moneyFormat.format(savingBalance));

}

else {

System.out.println("Balance cannot be Negatrive." +"\n");

}

}

} // end class BankAccount