ATM projesinde aldigim notlarin dokumani. Sonra ekle bunlari

package atmproject;

import java.text.DecimalFormat;

import java.util.Scanner;

public class Account {

private int accountNumber; // hesap no

private int pinNumber; // sifre

private double checkingBalance; // vadesiz hesap bakiyesi

private double savingBalance; // vadeli hesap bakiyesi

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

Scanner input = new Scanner(System.in);

public int getAccountNumber() {

return accountNumber;

}

public void setAccountNumber(int accountNumber) {

this.accountNumber = accountNumber;

}

public int getPinNumber() {

return pinNumber;

}

public void setPinNumber(int pinNumber) {

this.pinNumber = pinNumber;

}

public double getCheckingBalance() {

return checkingBalance;

}

public void setCheckingBalance(double checkingBalance) {

this.checkingBalance = checkingBalance;

}

public double getSavingBalance() {

return savingBalance;

}

public void setSavingBalance(double savingBalance) {

this.savingBalance = savingBalance;

}

public Scanner getInput() {

return input;

}

public void setInput(Scanner input) {

this.input = input;

}

// once diyelimki para cekme isleminde cikartma islemi yapmamiz lazim

// paraCekmeIslemindenSonraKalanMiktar amount= miktar

private double calculateCheckingBalanceAfterWithdraw(double amount){

checkingBalance = checkingBalance-amount;

return checkingBalance;

}

// para yatirma isleminden sonra kalan bakiyeyi hesapla

private double calculateCheckingBalanceAfterDeposit (double amount) {

checkingBalance= checkingBalance + amount;

return checkingBalance;

}

// para cekme : saving hesabindan para cekildikten sonra kalan bakiye

private double calculateSavingBalanceAfterWithdraw (double amount){

savingBalance = savingBalance - amount;

return savingBalance;

}

// saving hesabina para yatirdiktan sonraki bakiyeyi hesaplayiniz.

private double calculateSavingBalanceAfterDeposit(double amount){

savingBalance =savingBalance+amount;

return savingBalance;

}

// ikinci kisma geldik : Musteri ile para cekmek icin etkilesime gecelim: Checking hesabi icin

// bu method public olmali cunku baska classlardan kullanacagiz

// checking hesabinda paracekme durumu

public void getCheckingWithdraw(){

System.out.println("Checking Hesabinizda bulunan Bakiye: = "+ moneyFormat.format(checkingBalance));

System.out.println("Cekmek istediginiz miktari giriniz");

double amount = input.nextDouble();

if (amount<=0 ) {

System.out.println("Sifir yada eksi rakamlar gecersizdir!");

getCheckingWithdraw(); // recursive method (kendisini tekrardan cagirma)

}else if(amount <= checkingBalance){

calculateCheckingBalanceAfterWithdraw(amount); // yani parayı çekme başarılı ve sonrasında geri kalan miktarı döndürüyoruz müşteriye

// yukarida olusturdugumuz metodu buraya koyuyoruz.

// kisi hesabindan para cektikten sonra kalan miktari yazdirmak lazim

System.out.println("Checking Hesabinizda bulunan Bakiye: = "+ moneyFormat.format(checkingBalance));

}else {

System.out.println("Bakiyeniz yetersizdir");

}

}

// Para yatirma (checking hesabi icin) : Musterinin para yatirmasi icin hesaba gecelim.

private void getCheckingDeposit(){

System.out.println("Checking Hesabinizda bulunan Bakiye: = "+ moneyFormat.format(checkingBalance));

System.out.println("Yatirmak istediginiz miktari giriniz");

double amount = input.nextDouble();

if(amount<=0) {

System.out.println("Sifir yada eksi rakamlar gecersizdir!");

getCheckingDeposit();

}else {

calculateCheckingBalanceAfterDeposit(amount); // yani vermis oldugunuz rakami yatirabilirsiniz.

System.out.println();

System.out.println("Checking Hesabinizda bulunan Bakiye: = "+ moneyFormat.format(checkingBalance));

}

}

// son bakiyeyi gosteren metodu olusturuyoruz. cunku her zaman cagiriyoruz onu.

public void displayCurrenAmount(double balance){

System.out.println("Checking Hesabinizda bulunan Bakiye: = "+ moneyFormat.format(balance));

}

}

-----------------

package atmproject;

public class Runner {

public static void main(String[] args) {

Options options = new Options();

options.login();

}

}