**Name: ABDUL GHAFFAR KALHORO**

**Registration no: 194699**

**Class: BSCS\_6C**

**OBJECT ORIENTED PROGRAMMING**

**LAB\_12**

**Source Code:**

**TransactionRecord.java**

**public class TransactionRecord {**

**private int accountID;**

**private double amount;**

**public double getAmount(){**

**return amount;**

**}**

**public int getAccountNumber(){**

**return accountID;**

**}**

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

**this.amount = amount;**

**}**

**public void setAccountNumber(int accountID){**

**this.accountID = accountID;**

**}**

**}**

**AccountRecord.java**

**public class AccountRecord {**

**private int account;**

**private String firstName;**

**private String lastName;**

**private double balance;**

**private double totalValue;**

**public AccountRecord()**

**{**

**this(0,"","",0.0);**

**}**

**public AccountRecord(int acct, String first, String last, double bal)**

**{**

**setAccount(acct);**

**setFirstName(first);**

**setLastName(last);**

**setBalance(bal);**

**}**

**public AccountRecord(int acct, double trans){**

**setAccount(acct);**

**setBalance(trans);**

**}**

**public void setAccount(int acct){**

**account = acct;**

**}**

**public int getAccount(){**

**return account;**

**}**

**public void setFirstName(String first){**

**firstName = first;**

**}**

**public String getFirstName(){**

**return firstName;**

**}**

**public void setLastName(String last){**

**lastName = last;**

**}**

**public String getLastName(){**

**return lastName;**

**}**

**public void setBalance(double bal){**

**balance = bal;**

**}**

**public double getBalance(){**

**return balance;**

**}**

**public void total(TransactionRecord Trecord){**

**totalValue = balance + Trecord.getAmount();**

**}**

**public double gettotalValue(){**

**return totalValue;**

**}**

**}**

**Information.java**

**import java.io.FileNotFoundException;**

**import java.io.IOException;**

**import java.util.Formatter;**

**public class Information {**

**private Formatter fileInfo1;**

**private Formatter fileInfo2;**

**private AccountRecord AllInfo[];**

**private AccountRecord Recordtransaction[];**

**public Information() throws FileNotFoundException{**

**try{**

**fileInfo1 = new Formatter("oldmast.txt");**

**fileInfo2 = new Formatter("trans.txt");**

**}**

**catch(FileNotFoundException filenotfound){**

**System.err.format("%s", "This File can not be found");**

**System.exit(1);**

**}**

**catch(SecurityException authorization){**

**System.err.format("%s" , "This file is immutable");**

**System.exit(1);**

**}**

**AllInfo = new AccountRecord[4];**

**AllInfo[0] = new AccountRecord(100,"Alan", "Jones", 348.17);**

**AllInfo[1] = new AccountRecord(300,"Mary", "Smith", 27.19);**

**AllInfo[2] = new AccountRecord(500,"Sam", "Sharp", 0.00);**

**AllInfo[3] = new AccountRecord(700,"Suzy", "Green", -14.22);**

**Recordtransaction = new AccountRecord[4];**

**Recordtransaction[0] = new AccountRecord(100, 27.14);**

**Recordtransaction[1] = new AccountRecord(300, 62.11);**

**Recordtransaction[2] = new AccountRecord(400, 100.56);**

**Recordtransaction[3] = new AccountRecord(900, 82.17);**

**editFileInfo1();**

**editFileInfo2();**

**}**

**public void editFileInfo1(){**

**fileInfo1.format("%d %s %s %.2f%n", AllInfo[0].getAccount(), AllInfo[0].getFirstName(),AllInfo[0].getLastName(),AllInfo[0].getBalance());**

**fileInfo1.format("%d %s %s %.2f%n", AllInfo[1].getAccount(), AllInfo[1].getFirstName(),AllInfo[1].getLastName(),AllInfo[1].getBalance());**

**fileInfo1.format("%d %s %s %.2f%n", AllInfo[2].getAccount(), AllInfo[2].getFirstName(),AllInfo[2].getLastName(),AllInfo[2].getBalance());**

**fileInfo1.format("%d %s %s %.2f%n", AllInfo[3].getAccount(), AllInfo[3].getFirstName(),AllInfo[3].getLastName(),AllInfo[3].getBalance());**

**fileInfo1.close();**

**}**

**public void editFileInfo2(){**

**fileInfo2.format("%d %.2f %n", Recordtransaction[0].getAccount(), Recordtransaction[0].getBalance() );**

**fileInfo2.format("%d %.2f%n", Recordtransaction[1].getAccount(), Recordtransaction[1].getBalance());**

**fileInfo2.format("%d %.2f%n", Recordtransaction[2].getAccount(), Recordtransaction[2].getBalance());**

**fileInfo2.format("%d %.2f%n", Recordtransaction[3].getAccount(), Recordtransaction[3].getBalance());**

**fileInfo2.close();**

**}**

**public Formatter getFile1(){**

**return fileInfo1;**

**}**

**public Formatter getFile2(){**

**return fileInfo2;**

**}**

**public AccountRecord[] getAllInfo(){**

**return AllInfo;**

**}**

**public AccountRecord[] getRecordTransaction(){**

**return Recordtransaction;**

**}**

**}**

**MatchFile.java**

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Formatter;

import java.util.Scanner;

public class MatchFile {

private Scanner viewFile1;

private Scanner readFile2;

private Information Info;

private AccountRecord record;

private Formatter newMast,logFile;

public MatchFile() throws FileNotFoundException{

record = new AccountRecord();

Info = new Information();

newMast = new Formatter("newmast.txt");

logFile = new Formatter("log.txt");

try {

viewFile1 = new Scanner(new File("trans.txt"));

readFile2 = new Scanner(new File("oldmast.txt"));

} catch (FileNotFoundException ex) {

System.err.format("File cannot be found");

}

compare();

}

public void compare(){

for(AccountRecord master : Info.getAllInfo()){

for(AccountRecord trans : Info.getRecordTransaction()){

if(master.getAccount() == trans.getAccount()){

newMast.format("%d %.2f%n",master.getAccount(), master.getBalance() + trans.getBalance());

newMast.flush();

break;

}

else if(master.getAccount() != trans.getAccount() && checkTrans(master) == false){

newMast.format("%d %n" , master.getAccount());

newMast.flush();

break;

}

}

}

LogEdit();

}

public boolean checkTrans(AccountRecord master){

for(AccountRecord trans : Info.getRecordTransaction())

if(master.getAccount() == trans.getAccount())

return true;

return false;

}

public void LogEdit(){

for(AccountRecord transaction : Info.getRecordTransaction()){

boolean hold = false;

for(AccountRecord master : Info.getAllInfo())

if(transaction.getAccount() == master.getAccount())

hold = true;

if(hold == false){

logFile.format("Unmatched Transaction Record for account Number %d%n" , transaction.getAccount());

logFile.flush();

continue;

}

}

}

}

MainFile.java

import java.io.FileNotFoundException;

public class MainFile {

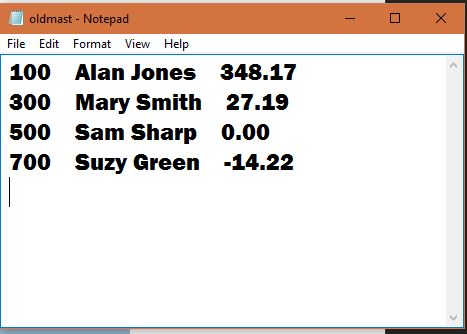
public static void main(String[] args) throws FileNotFoundException{

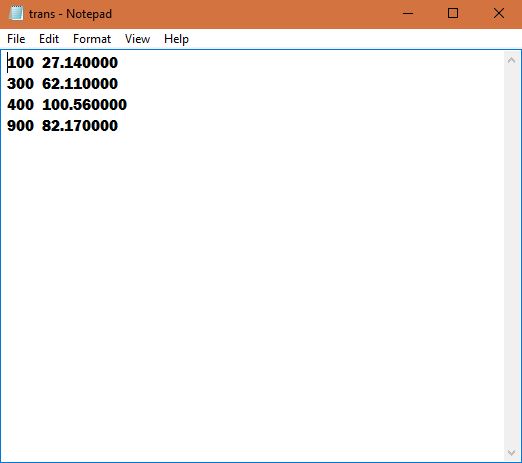
MatchFile objectfile = new MatchFile();

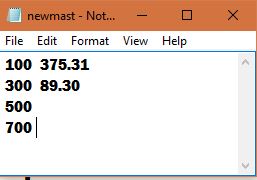
}

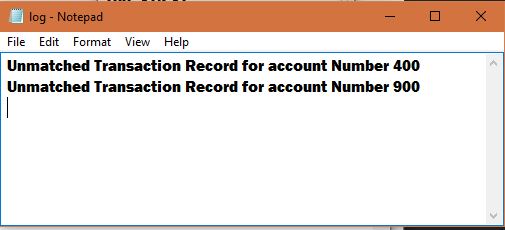
}

**OUTPUT**

****

****

****

****