1. A program that will settle bills among the people. For test data, used the below data

Bill 1 - Lunch

| **Total Amount** | **2000** |
| --- | --- |
| Paid By | Balaji |
| Shared by | Anand, Balaji, Chaitanya, Divya |

Bill 2 - Movie ticket

| **Total Amount** | **1000** |
| --- | --- |
| Paid By | Anand |
| Shared by | Anand, Balaji, Chaitanya, Divya |

Bill 3 - snacks (excludes Divya)

| **Total Amount** | **750** |
| --- | --- |
| Paid By | Chaitanya |
| Shared By | Anand, Balaji, Chaitanya |

The program should settle the bills among the people and print who pays how much to whom.

program

import java.util.\*;

public class BillSettlement {

public static void main(String[] args) {

Map<String, Integer> balances = new HashMap<>();

List<Bill> bills = new ArrayList<>();

// Add bills to the list

bills.add(new Bill("Lunch", 2000, "Balaji", Arrays.asList("Anand", "Balaji", "Chaitanya", "Divya")));

bills.add(new Bill("Movie ticket", 1000, "Anand", Arrays.asList("Anand", "Balaji", "Chaitanya", "Divya")));

bills.add(new Bill("Snacks", 500, "Chaitanya", Arrays.asList("Anand", "Balaji", "Chaitanya")));

// Calculate balances

for (Bill bill : bills) {

String paidBy = bill.getPaidBy();

int totalAmount = bill.getTotalAmount();

List<String> sharedBy = bill.getSharedBy();

int sharePerPerson = totalAmount / sharedBy.size();

// Add amount to paidBy person

if (!balances.containsKey(paidBy)) {

balances.put(paidBy, totalAmount);

} else {

balances.put(paidBy, balances.get(paidBy) + totalAmount);

}

// Subtract amount from sharedBy persons

for (String person : sharedBy) {

if (!person.equals(paidBy)) {

if (!balances.containsKey(person)) {

balances.put(person, -sharePerPerson);

} else {

balances.put(person, balances.get(person) - sharePerPerson);

}

}

}

}

// Print balances

for (Map.Entry<String, Integer> entry : balances.entrySet()) {

String person = entry.getKey();

int balance = entry.getValue();

System.out.println(person + " owes " + (-balance) + " to others.");

}

}

}

class Bill {

private String description;

private int totalAmount;

private String paidBy;

private List<String> sharedBy;

public Bill(String description, int totalAmount, String paidBy, List<String> sharedBy) {

this.description = description;

this.totalAmount = totalAmount;

this.paidBy = paidBy;

this.sharedBy = sharedBy;

}

public String getDescription() {

return description;

}

public int getTotalAmount() {

return totalAmount;

}

public String getPaidBy() {

return paidBy;

}

public List<String> getSharedBy() {

return sharedBy;

}

}

Summary:

This program calculates the balances owed between a group of friends after splitting bills. The program uses a Map to keep track of each person's balance, and a List of Bills to store the details of each transaction. For each bill, the program calculates the share per person and updates the balances accordingly. Finally, the program prints the balances owed by each person. The Bill class is used to store the details of each transaction.