Coding Assignment

The Problem

A group of people want to share expenses to keep track of spendings and who owes who what. You might know the apps "WeShare" or "Splitwise" – they solve the same problem.

An Example

John, Peter, and Mary go on a trip together. They want to keep track of expenses during their trip and settle all debts when the trip is over.

They share all expenses equally, but only one puts up money each time.

They also want to record when someone pays back another member of the group.

- On day 1, they arrive at the hotel and pay for it. It costs $500 for the three of them. John pays the hotel for them all.
- On day 2, they go out to eat. It costs them $150 in total. Mary pays at the restaurant.
- On day 3, they go sightseeing. It costs them $100 in total for various expenses that Peter pays for them all.

At the end of the trip, they have spent $750 in total.

Because they share everything equal, the trip has cost them $250 each.

John paid $500 for the hotel, so he should get $250 back from the two others.

Mary paid $150 at the restaurant, so she needs to pay $100 more.

Peter paid $100 for sightseeing, so he should pay $150 more.

In the end, Mary pays John $100 and Peter pays John $150. That way each person's total cost is $250.

It would be really nice, if they could have an app or web service that could help them with this!

Requirements

You should create the core business logic of an application that can help track expenses as exemplified above.

Functionality

The service should have the following functionality:

- an expense for the group can be added.
- a list of expenses can be obtained, so that everyone can know what has been spent and who put up money for it.
- the balance for each group member can be obtained.
- a payment from one member to another member of the group can be added.
- a list of payments that would settle all debts should be obtainable.

Assumptions

You may assume the following:

- The group may be fixed (e.g., hard-coded), so you do not have to worry about adding people to the group or removing people from the group.
- All expenses are shared equally among all members of the group.
- Only one person pay for each expense.
- All expenses are in the same currency, and there is only one currency.
- Nothing needs to be persisted – if the program stops, all data is flushed.

What to Implement

You should create:

- An interface with methods that provide the functionality.
- An implementation of that interface.
- Any classes used by the interface – for passing data, as helpers, etc.
- Some tests that can be run to demonstrate your work.

You do not need to implement anything else, e.g., web-interfaces, REST controllers, database-persistence are not needed for this (but we might talk about it).

So – a quite minimal implementation.