We have been asked to build a crowd-sourcing loan platform which will allow borrowers to take out loans that are made up of multiple loans from different lenders.
Users who want to borrow money (borrowers) will enter a loan request, which consists of the amount they want to borrow and the duration they want to borrow the money for. Users who want to lend money (lenders) will enter the amount of money they are willing to lend towards this loan and the interest rate (APR) they require.
The system should provide the following functionality via an API:

1. Create loan request – given an amount and duration, return an identifier for the loan request
2. Create loan offer – given a loan request identifier, amount and interest rate, return an identifier for the loan offer
3. Get current offer – given a loan request identifier, return the amount currently offered and the combined interest rate. The current offer should be the one providing the lowest combined interest rate. Partial loan offers can be used (i.e. if the total offered is greater than the loan request, the current offer should be capped at the loan request amount and can use part of a loan offer)

Example 1:
Given the following loan request:
Amount: 1000 | **Duration:** 100 days

And the following offers from lenders:

Amount	**Interest Rate (APR)**
100	5%
500	8.6%

The current offer should be:
Amount: 600
Interest Rate (APR): 8% (i.e. 100 @ 5%, 500 @ 8.6%)

Example 2:
Given the following loan request:
Amount: 1000 | **Duration:** 100 days

And the following offers from lenders:

Amount	**Interest Rate (APR)**
100	5%
600	6%
600	7%
500	8.2%

The current offer should be:
Amount: 1000
Interest Rate (APR): 6.2% (i.e. 100 @ 5%, 600 @ 6%, 300 @ 7%)

- Please implement the solution in Scala
- Your solution will be used as the basis for a pairing test where you will be asked to work on some UI functionality for this solution
- No persistence or UI is required; an in-memory solution providing an API is absolutely fine
- The system should support use by multiple users
- The solution should include everything you would check in to your source code management system
- The code should be of production quality
- Please provide a README with the following information:
  - How to run the application
  - How to run the tests
  - Reasons/justifications for each library or framework you decide to use, if any
    - Assumptions you made, if any, due to unclear requirements

Please zip up your entire solution and email it to your agent who will pass it on to us.