Quantexa - Coding Assignment

Introduction

The aim of this exercise is to test your programming ability. It is a take home assignment which has been sent to you before a potential 2nd round interview. If you are invited in for a second round interview we will discuss your solution to the problem.

The solution should be written Scala, as this is the primary language used at Quantexa and the solution should ideally be written in a functional style. You are able to use any functionality from the standard library in your solution. If you really don't want to write the solution in Scala, please get in touch to discuss alternative options.

Assignment

Data

You have been provided with a text file in comma separated format of 991 transactions spread over a month. The transactions are for multiple accounts and there are multiple types of transaction. The file has the following columns:

Field	Description
transactionId	String representing the id of a transaction
accountId	String representing the id of the account which made the transaction
transactionDay	Integer representing the day the transaction was made on (for simplicity we have removed any time information)
category	String representing the type of category of the transaction
transactionAmount	A double representing the value of the transaction

Questions:

Using the data provided, we would like you to answer the following 3 questions which requires calculating some statistics from the data. The output for each of the questions can either be provided as 3 files or the code can simply println the results to the console.

Question 1

Calculate the total transaction value for all transactions for each day.

The output should contain one line for each day and each line should include the day and the total value

Question 2

Calculate the average value of transactions per account for each type of transaction (there are seven in total).

The output should contain one line per account, each line should include the account id and the average value for each transaction type (ie 7 fields containing the average values).

Question 3

For each day, calculate statistics for each account number for the previous five days of transactions, not including transactions from the day statistics are being calculated for. For example, on day 10 you should consider only the transactions from days 5 to 9 (this is called a rolling time window of five days). The statistics we require to be calculated are:

- The maximum transaction value in the previous 5 days of transactions per account
- The average transaction value of the previous 5 days of transactions per account
- The total transaction value of transactions types “AA”, “CC” and “FF” in the previous 5 days per account

The output should contain one line per day per account id and each line should contain each of the calculated statistics, for example:

Day	Account ID	Maximum	Average	AA Total Value	CC Total Value	FF Total Value
9	A1	50	45.2	0	97	12
9	A2	400	122.2	1800	0	0
10	A1	50	44	17	92	11
10	A2	700	150	1600	100	0

Reading the file

The following code is an example in Scala of how to read a CSV file

```
  import scala.io.Source
  //Define a case class Transaction which represents a transaction
case class Transaction(     transactionId: String,     accountId:
String,     transactionDay: Int,     category: String,
transactionAmount: Double)

  //The full path to the file to import
  val fileName = "C:/Users/User1/Desktop/transactions.txt"
  //The lines of the CSV file (dropping the first to remove the header)
val transactionslines = Source.fromFile(fileName).getLines().drop(1)
  //Here we split each line up by commas and construct Transactions
val transactions: List[Transaction] = transactionslines.map { line =>
val split = line.split(',')
    Transaction(split(0), split(1), split(2).toInt, split(3), split(4).toDouble)
}.toList
```

The code above creates a List of transactions; this list can then be used to calculate the required statistics.