ME BANK TRANSACTION ANALYZER

Author: Tushar Sisode

Table of Contents

Key Solution Features:

- The solution has been built using core **JAVA 8** and uses only core **JAVA 8 APIs.**
- It does not use any frameworks like Spring, Hibernate or in-memory database.
- It heavily relies on **Java Stream API** and **Java Time API** for all the core operations.
- It uses **Gradle 5.4** as the build tool for compiling, packaging and running Junit test cases.

Solution Overview:

Solution is comprised of below core classes:

- **com.me.bank.TransactionsAnalyzer:** This is the main entry class (containing main method) of the application where the system is initialized, the user inputs are accepted and the output is displayed.
- **com.me.bank.model.Transaction:** This is the model class representing a Transaction record.
- **com.me.bank.mapper.TransactionMapper:** This is the mapper class for mapping the parsed transaction records to the Transaction model.
- **com.me.bank.engine.AnalysisEngine:** This is the engine class where the transactions are analyzed for the input account ID and the date range. It uses Java Stream API and Java Time API for querying and computing the output results.

Third Party Libraries:

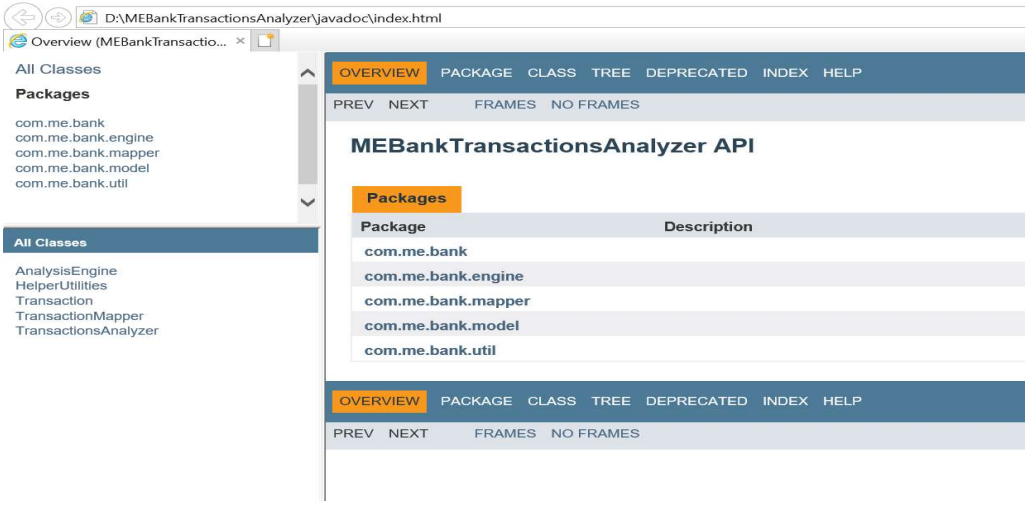
The solution uses only one third party dependency i.e. **Junit 4.12** which is used for unit testing.

Java API Docs:

The JAVA API Docs for the application are located inside folder **javadoc** under the root project folder.

```
D:\MEBankTransactionsAnalyzer>cd javadoc
```

Please open **index.html** in the browser to view the Java API docs as below:



Running the Project:

The application executable jar, **me-transactions-analyzer-1.0.0.jar**, is present inside folder **executable** under the root project folder.

Please run the application executable jar using cmd: **java -jar me-transactions-analyzer-1.0.0.jar**

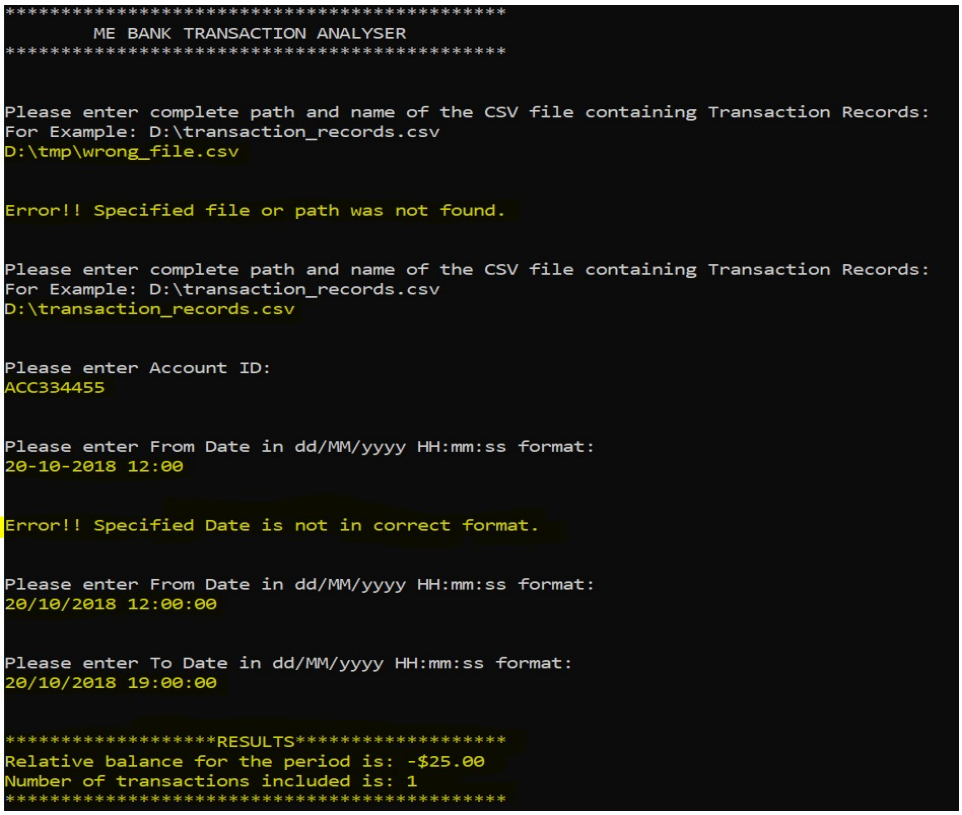
This looks like below:

```
D:\MEBankTransactionsAnalyzer\executable>java -jar me-transactions-analyzer-1.0.0.jar
```

The application runs in below steps:

- The application is initialized.
- Please enter **complete name and path of the CSV file** containing transaction records. The system validates the entered path and proceed only after the user enters the correct file path.
- Please enter the **account ID**.
- Please enter the **From Date**. The system validates if it is in correct format. It proceeds only after the date in entered in correct expected format.
- Please enter the **To Date**. The system validates if it is in correct format. It proceeds only after the date in entered in correct expected format.

The above steps are shown below:



Results for Different Input Sets:

- **Transaction records:**

  transactionId, fromAccountId, toAccountId, createdAt, amount, transactionType, relatedTransaction

  TX10001, ACC334455, ACC778899, 20/10/2018 12:47:55, 25.00, PAYMENT

  TX10002, ACC334455, ACC998877, 20/10/2018 17:33:43, 10.50, PAYMENT

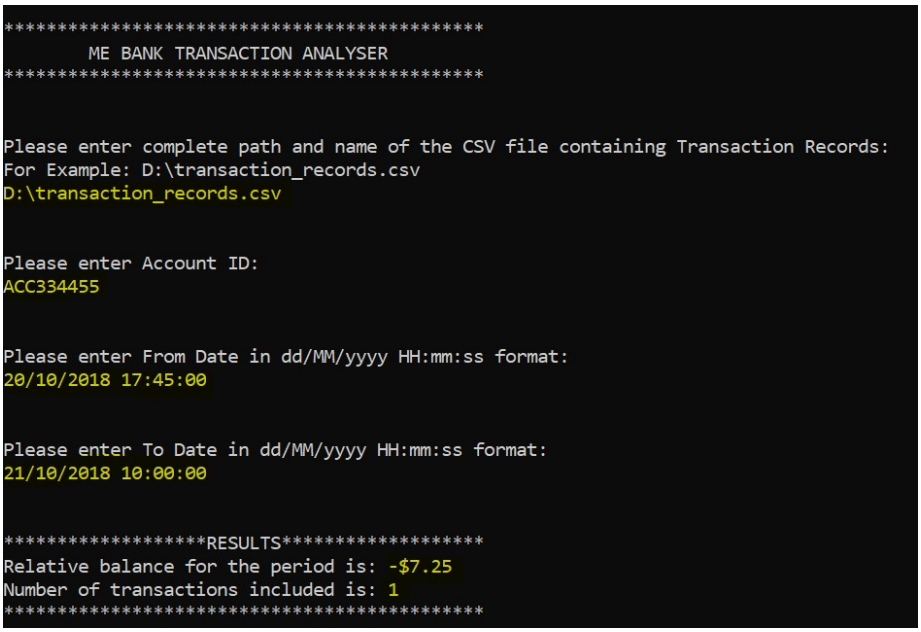
  TX10003, ACC998877, ACC778899, 20/10/2018 18:00:00, 5.00, PAYMENT

  TX10004, ACC334455, ACC998877, 20/10/2018 19:45:00, 10.50, REVERSAL, TX10002

  TX10005, ACC334455, ACC778899, 21/10/2018 09:30:00, 7.25, PAYMENT

- **Results:**

  Account ID: **ACC334455**, From Date: **20/10/2018 17:45:00**, To Date: **21/10/2018 10:00:00**



Account ID: **ACC998877**, From Date: **20/10/2018 17:00:00**, To Date: **20/10/2018 18:30:33**

```
*****************************************
        ME BANK TRANSACTION ANALYSER
*****************************************


Please enter complete path and name of the CSV file containing Transaction Records:
For Example: D:\transaction_records.csv
D:\transaction_records.csv


Please enter Account ID:
ACC998877


Please enter From Date in dd/MM/yyyy HH:mm:ss format:
20/10/2018 17:00:00


Please enter To Date in dd/MM/yyyy HH:mm:ss format:
20/10/2018 18:30:33


******************RESULTS******************
Relative balance for the period is: -$5.00
Number of transactions included is: 1
*****************************************
```

Account ID: **ACC778899**, From Date: **20/10/2018 12:00:00**, To Date: **21/10/2018 10:00:00**

```
*****************************************
        ME BANK TRANSACTION ANALYSER
*****************************************


Please enter complete path and name of the CSV file containing Transaction Records:
For Example: D:\transaction_records.csv
D:\transaction_records.csv


Please enter Account ID:
ACC778899


Please enter From Date in dd/MM/yyyy HH:mm:ss format:
20/10/2018 12:00:00


Please enter To Date in dd/MM/yyyy HH:mm:ss format:
21/10/2018 10:00:00


******************RESULTS******************
Relative balance for the period is: $37.25
Number of transactions included is: 3
*****************************************
```

Building the Project:

Please follow below build steps, if it is required to build the project for any reason.

- Navigate inside the project folder like below:

```
Microsoft Windows [Version 10.0.17134.648]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\vvq7dny>cd D:\MEBankTransactionsAnalyzer

C:\Users\vvq7dny>d:

D:\MEBankTransactionsAnalyzer>
```

- The project has been configured to be built with Gradle 5.4. Appropriate gradle wrapper has been setup inside the project root folder, which will automatically download the required gradle version upon executing the project build cmd.
- Build the project using cmd: **gradlew clean build**

```
Microsoft Windows [Version 10.0.17134.648]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\vvq7dny>cd D:\MEBankTransactionsAnalyzer

C:\Users\vvq7dny>d:

D:\MEBankTransactionsAnalyzer>gradlew clean build
```

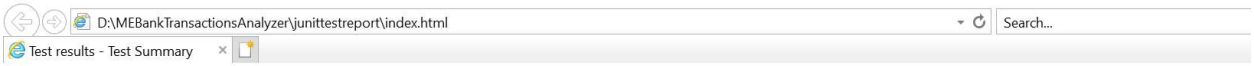
- If you are behind corporate proxy, set proxy configurations in gradle.properties under the root project before executing the build cmd as below:
  systemProp.http.proxyHost=www.somehost.org
  systemProp.http.proxyPort=8080
  systemProp.http.proxyUser=userid
  systemProp.http.proxyPassword=password
  systemProp.http.nonProxyHosts=*.nonproxyrepos.com|localhost
- Upon successful build completion, all the build artifacts are generated inside **build** folder under the project root folder. The application executable jar, **me-transactions-analyzer-1.0.0.jar**, is present inside path **build/libs**.

Junit Test Report:

The Junit Test Report for the application is located inside folder **junittestreport** under the root project folder.

```
D:\MEBankTransactionsAnalyzer>cd junittestreport
```

Please open **index.html** in the browser to view the Junit Test Report as below:



Test Summary

5	0	1	0.037s
tests	failures	ignored	duration

100%
successful

Ignored tests | Packages | Classes

Ignored tests

HelperUtilitiesTest. testingParseInputCsvFile

Packages

Package	Tests	Failures	Ignored	Duration	Success rate
com.me.bank.engine	1	0	0	0.028s	100%
com.me.bank.mapper	1	0	0	0.001s	100%
com.me.bank.util	3	0	1	0.008s	100%