Java developer test

Description:

Develop an application for parsing incoming data and converting the result of the parsing into the resulting format.

1. A simple task solution is required, as if the application could be supported and maintained by other less experienced developers.
2. The application must be implemented using the **Spring** framework.
3. The source code of the application must be a **maven** project and hosted on **GitHub**. Only dependencies from **public repositories** may be used.
4. The final application must be built by running the command:

**mvn clean install**

1. The application must be a console application. Example command to run: **java -jar orders\_parser.jar orders1.csv orders2.json**

where **orders1.csv** and **orders2.json** are parsing files.

1. The result of the execution should be output to the **stdout** stream.

**Note**: only output data should go into stdout, no logs should be there.

1. Parsing and conversion must be done in parallel in multiple threads.
2. Provision must be made for correct error handling in the source files. For example, a string value in the amount field may be instead of a number
3. Language tools no higher than **Java 8** may be used.
4. Consideration should be given to adding new formats for incoming data

**Input data**

1. CSV file.

Columns:

Order ID, amount, currency, comment

Example:

1,100,USD,order payment

2,123,EUR,order payment

Note: All columns are required

1. JSON file.

Example:

{"orderId":3, "amount":1.23, "currency": "USD", "comment": "order payment"}

{"orderId":4, "amount":1.24, "currency": "EUR", "comment": "order payment"}

Note: All fields are required

**Output data**

{“id”:1,“orderId”:1,”amount”:100,”comment”:”order payment”,”filename”:”orders.csv”,”line”:1,”result”:”OK”}

{“id”:2,“orderId”:2,”amount”:123,”comment”:”order payment”,”filename”:”orders.csv”,”line”:2,”result”:”OK”}

{“id”:3,“orderId”:3,”amount”:1.23,”comment”:”order payment”,”filename”:”orders.json”,”line”:1,”result”:”OK”}

{“id”:4,“orderId”:4,”amount”:1.24,”comment”:”order payment”,”filename”:”orders.json”,”line”:2,”result”:”OK”}

* id - order identifier
* amount - order amount
* currency - currency of the order amount
* comment - comment on the order
* filename - name of the source file
* line - line number of the source file
* result - result of parsing the source file entry
  + OK - if record was converted correctly,

or error description if the record was not converted correctly