Data for the project is included in the sample CSV file, attached. Please create a Github or Bitbucket repository for the project and push all code there; please email the link to the repository when you submit your project.

Customer X just informed us that we need to churn out a code enhancement ASAP for a new project. Here is what they need:

1. We need a Java application that will consume a CSV file, parse the data and insert to a SQLite In-Memory database.

a. Table X has 10 columns A, B, C, D, E, F, G, H, I, J which correspond with the CSV file column header names.

b. Include all DDL in submitted repository

c. Create your own SQLite DB

2. The data sets can be extremely large so be sure the processing is optimized with efficiency in mind.

3. Each record needs to be verified to contain the right number of data elements to match the columns.

a. Records that do not match the column count must be written to the bad-data-<timestamp>.csv file

b. Elements with commas will be double quoted

4. At the end of the process write statistics to a log file

a. # of records received

b. # of records successful

c. # of records failed

Rules and guidelines:

1) Feel free to use any online resources

2) Utilizing existing tools like Maven and open source libraries is encouraged.

3) A finished solution is great but if you do not get it all completed, that is ok - we will evaluate based on approach

4) It is required that you provide a README detailing the challenge. Your readme should include instructions to run your code, and a brief paragraph describing the approach you took to solve the challenge

Any technical questions regarding the challenge, please reach out to Matt Murgia at [mmurgia@ms3-inc.com](mailto:mmurgia@ms3-inc.com) or Josh Erney at [jerney@ms3-inc.com](mailto:jerney@ms3-inc.com).