**Automation Developer – Hands-on Exam**

**Background**

The date-parser project is a library that lets you parse different format of dates to “*java.util.Date”* format. The library can also convert text to dates.

Examples:

* “28-Feb-2010” will be converted to Sun Feb 28 10:32:26 IST 2010.
* “Next Monday” (if today is 3.10.2021) will be converted to Tue Oct 11 10:32:26 IDT 2021. (Supports the keyword “last” as well).
* “2020-04-20” will be converted to Mon Apr 20 11:32:41 IDT 2020
* “the day before tomorrow” will be converted to today’s date (Supports the keyword “after” as well).

The library uses ANTLR, which is a powerful parser generator for reading, processing, executing, or translating structured text or binary files. It's widely used to build languages, tools, and frameworks. From a grammar, ANTLR generates a parser that can build and walk parse trees.

Once building the project, based on the ANTLR files under “***src/main/antlr3/com/af/dateparser/generated***”, several JAVA classes will be generated automatically under “***src/main/java/com/af/dateparser/generated***”. These are standard JAVA classes, which contain the actual code that will be executed during runtime.

In addition, under “test” directory, there is a “Tester.java” file, which includes a few test cases.

Your mission, should you choose to accept it, is to complete the tester file.

**Task #1 – Test Coverage**

Build a satisfactory set of tests to cover the functionality. Pay attention to the followings:

* Test Coverage – cover as much functionality as you can
* Coding standards
* Code Cleanliness

**\* The tests should cover all cases based on the examples above!**

**\*\* No need to cover other functionalities of the date-parser library, only date parsing based on the above examples.**

**Task #2 – Bugs**

We’ve hidden a few bugs in the project. **Find** and **fix** them.

**Task #3 – Usage**

Under *src\main\resources* directory, you will find a csv file named “*stocks.csv*”, which contains the name, date, price, and volume of different stocks.

1. Create a new Java class with name “Task3.java”.
2. Load the CSV file and parse the data column with “date-parser” library.
3. In case there are two records or more with the same Symbol and the same Date, keep the record **with the highest price only!!!**
4. Sort the data by date, from oldest to newest.
5. Save the new filtered data into a new csv file named “updated-stocks.csv” under the same directory.

* Keep code complexity to a minimum.
* Keep code cleanliness