Write a program to accept a directory path and keep polling the path for new files. if it finds new files it should process those new files only . It should maintain a cache for file and that cache should be time based and configurable. The values can be kept in a property file; the cache should refresh itself based on that. Traverse the directory including all of it sub-directory find all the .txt and .csv files.

1. Find number of

i. Words

ii. Letters

iii. Vowels

iv. Special Characters (@, #, $, \*)

Write the details back into a file with the same name of the source file but the extension should be .mtd.

2. Create another file inside each directory with the same name of the directory with the extension .dmtd should contain the aggregated values of each .mtd file information.

3. Create another file inside each directory with the same name of the directory with the extension .smtd should contain the file names based on the sorted parameter passed when starting the application. Along with the value i.e. if sorting was based on words then it should display file name with total no of words in it.

3. The sorting order can be maintained in a property file and when cache refresh happens, it should read property file so see if sorting mechanism has been changed i.e. changed to vowels from words.

Note:

1. Use Threads & other synchronized package APIs.

2. Use collection frameworks and Design Patterns were ever its needed.

3. Write JUnit test cases for the functionality and code coverage.

4. Use java 8 API features.

5. The file processing should happen in parallel using multi-threading.