Write Into File

May 21, 2020

1 Algorithm

This code opens a file and writes some data

2 What to focus on

1. how COBOL defines a file
2. how COBOL defines the content of a file
3. notice how the different languages deal with opening and closing of files

3 Notes

The code defines the complex data type Employee. We recommend reading lesson 003 on complex data types. The best way of understanding the way data are ordered in a file is to picture a table or a database. You must define the structure of a row and then fill in the data. Our complex data type is just that: A row in a table. Thus Our file can be pictured like a database or table containing all of the data.
After that, notice that cobol does distinguish between data located on a file and data located in memory. To do so, it uses different sections under the DATA DIVISION. The FILE section lists the variables located on the file whereas the WORKING-STORAGE section list the data in the computer memory. The process of writing itself is very similar to other programming languages: Open the file, write the data, close the file.

4 Translation

As discussed in lesson 003, java and kotlin do not have data types similar to what cobol uses. As a consequence, we devised a stategy around it in lesson 003. Here, we just copy and paste those objects and refer the readers to those notes.
Concerning the operations on files, java and kotlin have different approaches. Kotlin's approach is simpler: The compiler takes care of opening and closing files, so you do not explicitly do that. In java, you must be more careful. The file itself is nothing more than a path, so nothing to open or close. In contrast, the fileWriter that writes into the file must be opened and closed. In addition, java requires to catch exceptions otherwise it does not compile.