Code Exercise

This is the code test for the job position of **Java Developer** at Ziath LTD. We ask you to implement a **Java** solution from the problem stated below. We will evaluate your solution on the bases of your adherence to Object-Oriented programming principles and good practices as well as code clarity.

Make sure to take advantage of any functionality already included in the standard Java library. You are free to use any external libraries if you feel you need to.

Domain Description

We need to implement a system that will manage biological samples (blood, tissue, and saliva) from different subjects. Each sample identified by an ID will have a volume (ml) and a sampling date.

Samples need to be stored 10 ºC or less, except from blood samples than need to be stored at 5 ºC or less. We need to store the blood type for blood samples (A, B, AB, O) and the organ of origin for the tissue samples. Samples are collected from Subjects which are identified by an ID and we have record of their name and surname.

Our system has operators that will process the samples. Each Subject is assigned a single operator that will be responsible for processing their samples. One operator can observe more than one Subject. Operators are identified by an ID, and their profile includes name and surname.

Samples are stored in tubes arranged in racks. Each rack is a grid of tube slots and each tube is assigned to a particular position (row and column) within a rack. The quantity of tubes in a rack determines its total capacity.

Data

System data is provided in the following CSV files:

- samples.csv
  - contains sample ID, sample type, volume, sampling date, subject ID, Rack ID, row and column in the rack, blood type (only for blood samples) and origin (only for tissue samples)
- personal.csv
  - contains personal information of both subjects and operators: ID, name, and surname and a boolean field indicating if the person is an operator.
- operators.csv
  - indicates which operator is observing each subject given by their ID's
- racks.csv
  - contains racks ID, number of rows and number of columns

Task

Create a Java application that implements the following functionality: The program must load the system data from the files referred above to memory and print to the screen the information indicated by each of the points below:

1. List all the Subjects. Including ID, name, and last name.
2. List all the Samples first blood, then tissue and then saliva. On each group order by date. Including sample type, ID, date, and volume.
3. List all the samples sorted by volume. Including sample type, ID, date, and volume.

4. List all samples of saliva type that are observed by the operator with ID 4. Including sample type, ID, date, and volume.
5. Find the operator who is responsible for processing sample 14. Show its ID, name, and surname.
6. List all the racks with their required storage temperature which is the minimum of all the storage temperatures for the samples within the rack. Include the ID of the rack, its total capacity, and the required storage temperature.
7. (Optional) We want to restructure our workforce to guarantee each operator is assign an adequate workload. Implement a heuristic to estimate the minimum number of operators we need to support all the subjects if each operator can only observe 5 samples

Submission

Please submit the source code of your solution to karem.terry@ziath.com and jini.thomas@ziath.com within three days of you receiving this document. The source code can be provided in a ZIP or in a public Git repository. Feel free to include any additional text with instructions on how to run the program.

Make sure to read carefully all the information provided in this document and the evaluation criteria. If you have any questions do not hesitate to ask via email to any of the addresses above.

Good luck!