Candidate Evaluation Criteria

Before you start, please check out the items below, they are the evaluation criteria that we will be using to evaluate your test.

* Clean Code
* Simplicity
* Logic
* SOC (Separation of Concerns)
* Flexibility/Extensibility
* Scalability/Performance

You can code the solution in any of the following languages: Java, scala, ruby, python, c#, javascript, clojure. The CODE needs be written into one of those languages:
if you submit the code in DELPHI for instance you will FAIL the test.

No question will be answered so you are free to solve the problem the way you think is best.

Statement of Work

You must build a data analysis system 100% coded in any of the languages above. The system must be able to import lots of flat files, read and analyse the data, and output a report.

Please read the following for more information about the input files, data analysis and the needed output.

Flat file layout

There are 3 kinds of data inside those files. For each kind of data there is a different layout.

Salesman data
Salesman data has the format id 001 and the line will have the following format.

```
001çCPFçNameçSalary
```

Customer data
Customer data has the format id 002 and the line will have the following format.

```
002çCNPJçNameçBusiness Area
```

Sales data
Sales data has the format id 003. Inside the sales row, there is the list of items, which is wrapped by square brackets []. The line will have the following format.

```
003çSale IDç[Item ID-Item Quantity-Item Price]çSalesman name
```

Sample file data

The following is a sample of the data that the application should be able to read. Note that this is a sample, real data could be 100% different.

```
001ç1234567891234çDiegoç50000
001ç3245678865434çRenatoç40000.99
002ç2345675434544345çJose da SilvaçRural
002ç2345675433444345çEduardo PereiraçRural
003ç10ç[1-10-100,2-30-2.50,3-40-3.10]çDiego

003ç08ç[1-34-10,2-33-1.50,3-40-0.10]çRenato
```

Data analysis

Your system must read data from the default directory, located at %HOMEPATH%/data/in. The system must only read .dat files.

After processing all files inside the input default directory, the system must create a flat file inside the default output directory, located at %HOMEPATH%/data/out. The filename must follow this pattern, {flat_file_name}.done.dat.

The output file contents should summarize the following data:

- Amount of clients in the input file
- Amount of salesman in the input file
- ID of the most expensive sale
- Worst salesman ever

This application should be running all the time, without any breaks. Everytime new files become available, everything should be executed.

Application Construction

As long as your code is written in one of the languages above, you are free to build whatever kind of application you feel is suitable for the job.

You have total freedom to google everything you need. Feel free to pick any external library if you need so.

Just keep in mind the criterias above. Good luck :)