# Engineering Exercise – Clover Health

## Instructions

Part of the onsite includes a portion where we want to evaluate your code. Please complete this exercise and get it back to us within 48 hours of your visit. You should only spend about 2-3 hours of your time and is a way for our engineers to dive in your work during the coding review panel.

## Problem definition

You have directories containing data files and specification files. The specification files describe the structure of the data files. Write an application in the language of your choice that reads format definitions from specification files. Use these definitions to load the data files into a database.

## Problem details

Data files exist in a data/ directory relative to your application and specification files exist in a specs/ directory relative to your application.

Specification files will have filenames equal to the file type they specify and extension of .csv. So fileformat1.csv would be the specification for files of type fileformat1.

Data files will have filenames based on their specification file name, followed by an underscore, followed by the drop date and an extension of .txt. For example, fileformat1\_2007-10-15.txt would be a data file to be parsed using specs/fileformat1.csv, which arrived on 10/15/2007.

Format files will be csv formated with columns "column name", "width", and "datatype".

* "column name" will be the name of that column in the database table
* "width" is the number of characters taken up by the column in the data file
* "datatype" is the SQL data type that should be used to store the value in the database table.

Data files will be flat text files with lines matching single records for the database. Lines are formatted as specified by their associated specification file.

## Example

This is an example file pair; other files may vary in structure while still fitting the structure of the problem details (above):

specs/testformat1.csv:

"column name",width,datatype  
name,10,TEXT  
valid,1,BOOLEAN  
count,3,INTEGER

Here we have a specification that describes 3 columns:

* The first 10 characters labeled "name" of type TEXT
* The next 1 character labeled "valid" of type BOOLEAN ('1' = True, '0' = False)
* The last 3 characters labeled "count" of type INTEGER

data/testformat1\_2015-06-28.txt:

Foonyor   1  1  
Barzane   0-12  
Quuxitude 1103

Processing this data file results in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | name | valid | count |
| 1 | Foonyor | TRUE | 1 |
| 2 | Barzane | FALSE | -12 |
| 3 | Quuxitude | TRUE | 103 |

## Expectations

* Clover primarily uses Python, but your application can be written with language/libraries of your choosing. Take this opportunity to best demonstrate your talents!
* Database type and connection mechanism is left to your discretion.
* You should include tests that cover, at least, the examples given.
* You should implement the conversions for the SQL data types: TEXT, BOOLEAN, and INTEGER
* You should be able to handle any specification file that matches the problem description (not just the given example)
* Files can be assumed to use UTF-8 encoding
* You should be prepared to discuss implementation decisions and possible extensions to your application.