DATA QUALITY TOOL FOR SCHEMA INFERENCE AND COMPARISON

System Description:

Our data ingestion from native applications to Cloud-based Data Lake platform approximately follows the following procedure:

STEP 1: Business Analyst collects the Data Requirements from Business Users and creates a Data Delivery Agreement (DDA, .xlsx format) document which contains various data details including column names and column data types for all datasets from a particular source. This DDA is converted into .JSON files using a seperate in-house automated application which is then monitored for various issues. Recently, a new format of these .JSON have been introduced where there will be individual .JSON files per dataset

STEP 2: Application IT Team will now generate data from the application (for eg, Market Data API to collect public market information such as stock prices) as agreed on the DDA and send this data to the FTP Server. If the source intends to send 5 datasources, csv files corresponding all datasets are zipped into a .tar file and sent to the FTP. The correct filename pattern is used to detect which csv corresponds to which dataset.

STEP 3: Batch scheduling jobs are executed by Control-M, an application workflow orchestration platform, which sends the files in FTP to Staging area of the Cloud Server.

STEP 4: The files in the Staging Area are pulled into a particular Dataset object on the Data Lake. This final receiving dataset has a pySpark compatible schema.

Problem Description:

We have many encountered situations where data in the csv files sent from application (especially when there are many datasets, with many columns) have errors.

One way to identify errors is by monitoring the type of data in the csv files and comparing it with the DDA document (initial requirement).

Some eg. errors:

- According to DDA, the column closing_trade must be TimeStamp Type. But due to a miscommunication or sourcing wrong info, the values under this column come from an integer flag and so in the csv file happens to have values 0/1

- DDA says column Next_Review_Date must be Date Type. But in the csv file there are string values '-' as null fillers

- There are 13 columns in a dataset in the csv file where as the DDA says there should only be 12.

WHAT DO WE EXPECT FROM YOU?

I We want you to create a .py program can be run via command line

Inputs:

- File location: Please create a folder on your PC/cloud location of your choice and your program must be able to access the files in this location when the location is provided as an input. We will assume in the beginning that all concerning input files are stored at this location which will be accessed programmatically
  Note: We can later change this to enable accessing FTP location which would require authentication.

- DDA files (.JSON): There are two types of JSON files:
  (1) One type contains the names of all the datasets and overall parameters concerning these datasets. [Attachments]
  (2) Second, there are individual JSON files for each of the dataset (dataset name in the filename) which contains the schema (column names and datatypes) [Attachments]

Please find attached JSON files for your reference.

Please go through the JSON files carefully and understand what parameters are available. Your program must access the main JSON (1) and detect the names of the individual dataset names. Then with these names, you search for the individual JSON files. If any dataset is missing, please raise a warning but continue to schema inference of the datasets for which the individual JSONs are available.

- Dataset CSV files (Zipped folder, .tar):
  Given the required schemas in the JSON files, please create your own csv files for the same [one dataset in one csv file]. The name of the file should be as per the filename pattern given the main JSON (1).

  Please generate the sample data in your .csv files with as many different types of schema errors (like the examples provided above) such that you may test your program thoroughly. Once your have all your csv files, please zip them into a .tar file and also keep them the file location decided earlier.

PROGRAM:

- Program must ask user for input file location.
- Then the program scans through the files for the main DDA .JSON file
- Then the program gets the names of expected datasets.
- Then program will look for individual dataset JSONs to get the expected schema
- Then program unzips the tar folder and goes through each dataset csv file and detects if there are any possible errors.
- The program outputs a log - with WARNINGS about missing datasets and a list of ERROR/NO ERROR for each dataset. You may get creative here on what the output log will show.

II Using ChatGPT
Kindly leverage ChatGPT while working on this project (at every step as you wish and in as many creative ways). Make notes (including chat transcripts) on where it was really useful, where it wasn't, where it gave direct solutions and where you had correct the work a lot.

III Project Report on the Development & Testing
In your final project report, kindly elaborate on every functionality of your program, including method selection, assumptions and limitations.
Please also in detail explain how you have tested the program (with as many different posisble errors)
Please also describe (in a story format would be best) your journey with ChatGPT.