Technical Challenge: PDF Data Extraction and Processing

Purpose of the Exercise

Your task is to create a process in Python to extract specific data from a PDF document that we will provide. Then, you must organize this data according to the formats described below.

Development Guidelines

Your solution must meet the following requirements:

Tools and Language

- You must use Python as the main programming language.
- You can use any Python libraries you prefer to work with PDFs (e.g., PyPDF2, pdfplumber, Camelot, tabula-py, etc.).
- It is recommended to use Pandas and/or NumPy to organize and manipulate the extracted data.

Data Extraction and Transformation

Find and extract important tables or data from the PDF. Specifically, extract and process the data from **pages 2 and 3** of the PDF, and save it in a CSV or Excel file (.csv or .xlsx). Additionally, it will be highly valued if you also extract the data from **pages 16 to 19**. The output for these pages should have the same quality and structure as the data from the initial pages.

From the data on **page 2 ("Markets at a glance")**, consolidate all performance metrics (1M, 3M, 6M, 12M, YTD, QTD) from the Equities, Rates, Credit, Commodities, and Exchange Rates tables into a single structure. Then, identify and extract the names and 12M returns of the **top 3 and bottom 3 performing assets/indices** from the entire page. Clean and transform the extracted data to ensure it is consistent and correctly formatted (e.g., convert data types, handle null values, standardize column names). If useful, you may perform basic calculations or aggregations to show your understanding of the data (e.g., calculate averages).

Saving the Information

The data must be exported to a structured format — a **CSV or Excel file is preferred**. Make sure the final file is clean, organized, and easy to read.

Bonus: It is highly valued if, in addition, you store the data in a relational database that you set up yourself. This database should run inside a **Docker container**.

Repository and Documentation

We expect a repository with a clear and meaningful commit history. The project structure should be logical and easy to navigate.

Include a README.md file in the root of your repository that explains:

- How to run your solution.
- Any external libraries you used and how to install them (for example, using pip install -r requirements.txt).
- Any assumptions or challenges you encountered with the PDF or data extraction.
- A brief explanation of your approach and technical decisions.

Additional Valuable (Optional) Considerations

1. **Container for the Process:** It is a plus if you create a Docker container that includes your Python scripts and the PDF, so the process can run in an isolated, dockerized environment.
2. **Container Communication:** If you set up a database, it is also valued if you demonstrate communication between the two containers (one for the database and one for the Python process).

Estimated Time

This challenge is estimated to take **between 8 and 16 hours** to complete.

When you're done, please email your results to amarchi@focus-economics.com and cvives@focus-economics.com.

Thanks again for your collaboration and best of luck with the test!

Best regards,

FocusEconomics