READ\_ME PPAR Project Lessons (Databricks)

**Overview:**  
  
This code is designed to facilitate data processing and extraction from PPAR (Project Performance Assessment Report) files. It imports essential modules for file operations, regular expressions, time-related tasks, and data manipulation. The code includes functions to create a data folder, clean text data, perform bulk file download, process PPAR files, extract lessons, and prepare output data. It also provides helper functions for text manipulation and formatting. The code aims to streamline the extraction and organization of information from PPAR files, enabling further analysis or presentation of the extracted lessons and document details.

**Running the Code:**

1. Set Up Databricks Environment:

Log in to your Databricks workspace.

Create a new notebook or open an existing one where you want to run the script.

1. Install Required Libraries:

Ensure libraries like pandas and any custom libraries (APIDocInfo, bulkDownload) are installed and accessible in your environment.

1. Upload the Python Script:

Copy the Python script you have.

Paste the script into a cell (or multiple cells) in the Databricks notebook.

1. Upload Excel File with Project IDs:

Set up widgets in the Databricks notebook for user inputs (file\_path, output\_path, download\_version, excel\_path, and lending\_path) Excel\_path is only needed if you have to download PPAR files aswell.

Ensure your data (project files and Excel sheets) is uploaded to a location accessible by the Databricks cluster (like DBFS or mounted storage).

1. Run the Script:

Execute the notebook cells sequentially.

Monitor the execution for any errors and debug as necessary.

1. Fill the following empty fields:

* Folder Path for the downloaded files
* Folder Path for the final file (output)
* Excel Path for Lending.xlxs file

1. Run the code. If files are already downloaded (PPAR), then skip the commented two code chunk, if files are not downloaded, then download the PPAR files using the project\_ids by specifying the project\_id stored in a excel\_path, and the download version. These two options can be set by filling in the empty boxes. Once done just run the remaining part.
2. Access and Review Output:

Once the script completes, the output will be Excel files containing the extracted data and reports. These files will be saved to the specified output paths.