Data Engineering Pipeline Challenge

# Overview

The data analysis team requires a new data pipeline for downloading and cleaning energy trend data. The data, which is available on the UK government's website, contains information about the supply and use of various energy sources. Your job is to develop a script that can automatically download and clean this data, validate its quality, and save it as a CSV file. The script should be scheduled to run daily to check for any new datasets, ensuring that the latest data is always available to the analyst.

The analyst want you to start with the dataset "Supply and use of crude oil, natural gas liquids, and feedstocks" from the UK government's website: [https://www.gov.uk/government/statistics/oil-and-oil-](https://www.gov.uk/government/statistics/oil-and-oil-products-section-3-energy-trends) [products-section-3-energy-trends.](https://www.gov.uk/government/statistics/oil-and-oil-products-section-3-energy-trends) The data comes as an Excel spreadsheet with multiple tabs, where

the analyst is only interested in the “Quarter” tab.

Your task is to perform the following operations:

1. Write a script that can check for new data, and if a new dataset is detected. Download the new Excel file.
2. Clean the data to remove any unnecessary information and to ensure that the data is in a consistent and well-structured format.
   1. Missing values should be left blank
   2. Ensure that any dates and timestamps are converted into a standard dateformat of yyyy-MM-dd and yyyy-MM-dd HH:mm:ss for timestamps
   3. Retain information about when the data was processed and the original filename
3. Validate the schema by checking that the data is in the same format and contains the same information as the previous dataset that has been downloaded.
4. Save the resulting DataFrame to a CSV file in a format that can be easily ingested into a data lake.

# Technical Requirements

1. Must be implemented as a Python PIP Package, using Python greater than or equal to V3.7.
   1. If you are using Spark through a Notebook interface the Notebook should import the package
2. Must be implemented in either Pandas or PySpark. When using PySpark the package will be tested in a Databricks environment.
3. The location of the CSV should be as a parameter to the package.