**User Manual:**

**Wise Regulatory Data Processor**

**1. Introduction**

This Jupyter Notebook automates the reporting of Wise transaction data to regulatory bodies in the UK and US. It extracts data from an Excel file, transforms it to meet reporting requirements, and loads it into BigQuery for analysis.

**2. Prerequisites**

* Python 3.7 or higher
* Google Cloud Project with BigQuery
* Google Cloud Storage bucket
* Required Python libraries: pandas, openpyxl, google-cloud-storage, google-cloud-bigquery
* Wise transaction data in Excel format ("Dummy Data.xlsx")
* 3. Installation

Install the necessary Python libraries:

Bash

*pip install pandas openpyxl google-cloud-storage google-cloud-bigquery*

Set up your Google Cloud credentials. The easiest way is to set the GOOGLE\_APPLICATION\_CREDENTIALS environment variable:

Bash

export GOOGLE\_APPLICATION\_CREDENTIALS="/path/to/your/credentials.json"

**4. Usage**

**4.1 Data Loading**

* The notebook loads customer and transaction data from "Dummy Data.xlsx".
* Modify the excel\_file\_path variable if your data file has a different name or location.

**4.2 Data Cleaning**

* The clean\_data function performs the following cleaning steps:

Standardizes column names.

Converts date columns to datetime format.

Cleans up currency route information.

Converts the transaction amount to numeric.

The display\_dataframe\_info function displays the cleaned DataFrames.

**4.3 Data Quality Checks**

* The perform\_qa\_checks function performs basic data quality checks, including:

Checking the structure of the DataFrames.

Checking for null values and duplicates.

Validating date formats.

Checking referential integrity between customer and transaction data.

* The generate\_report function generates a data quality report in an Excel file named "data\_quality\_report.xlsx".

**4.4 Data Saving and Uploading**

* The save\_data function saves the cleaned DataFrames as CSV files.
* It uploads the CSV files to the specified Google Cloud Storage bucket.
* It creates or updates tables in BigQuery with the uploaded data.
* It uploads the data quality report to the specified Google Cloud Storage bucket.

**5. Configuration**

Update the following variables in the notebook:

bucket\_name: Name of your Google Cloud Storage bucket.

project\_id: Your Google Cloud project ID.

dataset\_id: BigQuery dataset ID.

table\_id\_customer: BigQuery table ID for customer data.

table\_id\_transactions: BigQuery table ID for transaction data.

destination\_blob\_name\_customer: Filename for the customer data on Google Cloud Storage.

destination\_blob\_name\_transactions: Filename for the transaction data on Google Cloud Storage.

**6. Running the Notebook**

Run all cells in the notebook sequentially. The notebook will:

1. Load data from Excel.
2. Clean and preprocess the data.
3. Perform data quality checks.
4. Save the cleaned data to CSV files.
5. Upload the CSV files to Google Cloud Storage.
6. Create and load tables in BigQuery.
7. Upload the data quality report to Google Cloud Storage.

**7. Troubleshooting**

* Refer to the log messages for detailed information about any errors encountered.
* Ensure that your Google Cloud credentials are set up correctly.
* Check the format and contents of your Excel data file.

**8. Support**

Contact your administrator or support team for any questions or assistance.