Guide for using ecs_olap repository

Introduction

https://github.com/heymarjay/ecs_olap has 4 subfolders named accordingly to their sequence:

a. 0_scrape
b. 1_data_model
c. 2_ssis
d. 3_output

0_scrape

1. These codes will scrape the latest data from https://database.coffeeinstitute.org/. The data from https://github.com/jldbc/coffee-quality-database/tree/master/data was last updated two years ago. You will need these libraries to run the codes successfully:
   a. selenium
   b. beautifulsoup4
   c. pandas
   d. lxml
   e. numpy

2. Run the codes sequentially, (hint: Like the folders, listing the codes in ascending order is a great way to indicate the order to run them.):
   a. 0000_ecs_extract_cqisite_robusta.py
   b. 0010_ecs_consolidate_raw_robusta.py
   c. 0100_ecs_extract_cqisite_arabica.py
   d. 0110_ecs_consolidate_raw_arabica.py

1_data_model

1. Moving forward, majority of the codes will be sql which has to be run in MS SQL Server. The codes in this folder will create databases, tables, and other necessary objects to build the OLAP schema and run the data pipeline successfully. You can find the table schema here: https://dbdiagram.io/d/5effe14b0425da461f043afa
2. Create the databases using the codes in 0_databases. There will be 2 databases: ECS_Transform for staging, and ECS for the output. Note that you might need to modify the installation path for the DB files.
3. Create the tables using codes in 1_tables folder. If this step fails, it might mean you skipped creating the databases. The prefix d_ denotes it is a dimension table, or f_ if it is a fact table.
4. Create the views using codes in 2_views folder. Dashboards and reports would pull data from views instead of getting it straight from tables. Views are prefixed with v_.
5. Create logs table using the code in 3_logs. The logs will maintain a record of operations of the entire data pipeline, which may help troubleshooting and monitoring performance.

2_ssis

1. The codes in these folders are best implemented in SSIS as sequence containers represented by the folders structure.
2. The codes will need to run sequentially. Note that for 2001_ecs_csv_load_to_staging, you will need to import the data from flat file cqidb_2020.csv using SSIS, SSMS, or bulk insert. SSIS is preferred for automation.
3. Codes in 1_clean_transform will treat and the data, run the codes sequentially:
   These codes will delete records which does not have primary key according to schema:
   a. 2100_delete_null_owner.sql
   b. 2101_delete_null_species.sql
   c. 2102_delete_null_country.sql

   These codes will transform the data which will make them easier to analyze:

   d. 2103_cast_facts.sql
   e. 2104_cast_altitude.sql
   f. 2105_clean_defects.sql
4. Run the codes in 2_staging which truncates the staging tables and inserts the data
5. Sequentially run the codes in 3_load, the flow is generally update the data if it exists and if the value has been changed. If the data does not exist in the final table, then it will insert the new data. All operations here are recorded in logs_table.

3_output

1. There are no codes here, just sample outputs. The excel spreadsheet has been extracted from v_all_scores The twbx file is a Tableau packaged workbook containing the sheet and dashboard. Install Tableau Desktop or Tableau Reader to open it.