**Basic ETL Project**

This project demonstrates a foundational understanding of ETL (Extract, Transform, Load) processes using Python, Pandas, and SQLite.

**Repository Contents**

1. **“data” Directory** 
   1. **Bank\_transactions\_data\_2.csv**
      1. Dataset sourced from Kaggle that can be found using the following [link](https://www.kaggle.com/datasets/valakhorasani/bank-transaction-dataset-for-fraud-detection).
      2. Intended for fraud detection model development.
      3. Used as the raw input for the ETL process.
   2. **staging\_customers.csv**
      1. File created when running the extraction step.
      2. Serves as the staging area for cleaning and validated data
   3. **project.db**
      1. SQLite database file created when running the full ETL pipeline.
      2. Stores all transformed and aggregated data tables
2. **“etl” Directory**
   1. **Extract.py**
      1. Extracts and validates raw transaction data from a specified CSV file by ensuring all required columns exist and removing duplicate transactions
      2. Stages the cleaned dataset by saving it to a designated staging file path and logs the process and any errors encountered
   2. **Transform.py**
      1. Transforms and cleans the extracted transaction data by converting date fields, created new time-based variables (month, day, hour, weekday), and removing invalid or unnecessary columns
      2. Aggregates transaction amounts across multiple dimensions (by month/day, hour and location) to produce summary DataFrames for further analysis
   3. **Load.py**
      1. Loads transformed DataFrames into a SQLite database by writing each DataFrame to its corresponding table within a single project database file
      2. Implements basic error logging and validation to ensure the number of DataFrames matches the expected table names before saving
   4. **run\_etl.py** 
      1. Runs extract.py, transform.py, and load.py
   5. **“logs” Directory** 
      1. Contains the extract.log file which should keep a track of any logging throughout the etl process

**How to Run**

1. Clone the repository
2. Verify that CSV file (bank\_transactions\_data\_2.csv) is located in the data/ directory
3. Navigate to the etl/ directory.
4. Run the ETL pipeline with “python run\_etl.py”