Data Engineering Test

Exercise 1: Data Ingestion using Web3.py

Instructions:

Provide a code that can perform the following request:

1. Using web3.py, connect to the Ethereum network (you can use a public node like Infura).
2. Extract transactions from a specific range of blocks
3. Save those that meet a specific condition into a JSON file:
   a. Condition: The 'to' field of the transaction matches a specific wallet address (e.g., '0xYourTargetWalletAddress')
   b. Save each matching transaction as a separate JSON file in a designated directory.

Exercise 2: Data Transformation

Instructions:

Using the [following data example](#) (may be extended if needed), write an SQL query to obtain the daily balances of each wallet in each token valued

in ETH using FIFO valuation criteria. The result should show the cumulative balance by day, by wallet, and by token in ETH.

Exercise 3: Data Modeling (Data Warehouse) for DeFi Transactions Reporting

Context:

You are working for a decentralised finance (DeFi) platform that allows users to perform various financial activities such as lending, borrowing, and trading cryptocurrencies.

The platform wants to set up a data warehouse to analyse transaction performance, user behaviour, and asset trends.

The data warehouse will be used by analysts and managers to generate reports and dashboards for decision-making.

Instructions:

1. Provide a detailed datawarehouse structure with definition of the tables, including column names, data types, and primary/foreign keys.
2. Ensure the schema supports queries for:
   a. Total transaction volume by asset, transaction type, and date.
   b. User activity and transaction history.
   c. Trends of different assets over time.
   d. Performance of different DeFi protocols.

Please report assumptions taken for the construction of the data warehouse.