Data Engineering Assignment

The Task

Implement a system to fetch and store stock price data and apply logic to buy or sell stocks under certain conditions.

The first run of the process should backfill 2 years of data. Subsequent runs of the process should import all new prices since the last run.

Multiple runs on the same date should be idempotent. Thus if the script is run multiple times on the same day, there will only be one closing price per stock per date in the final `prices` table. Similarly there should not be duplicated buy/sell orders in the final `orders` table.

Where relevant write tests for your code.

Your code should be able to handle errors appropriately.

You may use any language and any of the following databases: SQLite, MySQL, PostgreSQL, Redshift or BigQuery.

Downloading and Storing Data

- You will be supplied with a list of stock codes to use for this exercise, specified in the file `stock_codes.txt`. Codes will be newline separated.
- Use the free and open API provided by IEX Trading to download the data for the `prices` and `company` tables. You may find the `batch` endpoint useful for requesting for data for a batch of stock codes for any other endpoint.
- The process should import any new prices since the previous price import. For the first run the process should backfill 2 years of historic data.
- You do not need to update existing data.
- The following two data entities need to be downloaded and stored:

    a. `prices`
        i. API Endpoint: `chart`
        ii. Table fields: `stock_code`, `date`, `close`

    b. `company`
        i. API endpoint: `company`
        ii. Table fields: `company_name`, `exchange`, `sector`, `industry`

Apply trading strategies

- Directly after storing new price data, apply the Golden Cross trading strategy for all new prices and insert new orders to buy/sell that stock into an `orders` table.

- The Golden Cross strategy compares the short term moving average (SMA) of a stock price with the long term moving average (LMA).
  - When the SMA crosses the LMA in an upward direction then you must create a 'buy' order (because we believe that the market is trending up).
  - When the SMA crosses the LMA in a downward direction then you must create a 'sell' order.
  - If there is insufficient data, or the SMA and LMA are equal, create a 'hold' order.

- Use 45 days for the SMA and 180 for the LMA.

- The `orders` table should have the following fields
  - `stock_code`, `order_date`, `order_type`, `trading_strategy` (in this case 'Simple Golden Cross' but make provision for the fact that multiple types of trading strategies)

Deliverable

Please provide a link to git repo that contains the following:

- Scripts to setup and run the prescribed sequence of events (i.e. downloading and storing the data, creating orders etc.).
- Setup scripts to create any tables/views that your database requires.
- Instructions and, where necessary, scripts to perform any necessary setup and to run the scripts.
- Make sure to expose configuration options (where applicable) for any values that need to be modified in order to run the scripts in an environment that is different from yours.