BEES Data Engineering – Breweries Case

Objective:

The goal of this test is to assess your skills in consuming data from an API, transforming and persisting it into a data lake following the medallion architecture with three layers: raw data, curated data partitioned by location, and an analytical aggregated layer.

Instructions:

1. **API:** Use the Open Brewery DB API to fetch data. The API has an endpoint for listing breweries: <https://www.openbrewerydb.org/>
2. **Orchestration Tool:** Choose the orchestration tool of your preference (Airflow, Luigi, Mage etc.) to build a data pipeline. We're interested in seeing your ability to handle scheduling, retries, and error handling in the pipeline.
3. **Language:** Use the language of your preference for the requests and data transformation. Please include test cases for your code. Python and PySpark are preferred but not mandatory.
4. **Containerization:** If you use Docker or Kubernetes for modularization, you'll earn extra points.
5. **Data Lake Architecture:** Your data lake must follow the medallion architecture having a bronze, silver, and gold layer:
    a. **Bronze Layer:** Persist the raw data from the API in its native format or any format you find suitable.
    b. **Silver Layer:** Transform the data to a columnar storage format such as parquet or delta, and partition it by brewery location. Please explain any other transformations you perform.
    c. **Gold Layer:** Create an aggregated view with the quantity of breweries per type and location.
6. **Monitoring/Alerting:** Describe how you would implement a monitoring and alerting process for this pipeline. Consider data quality issues, pipeline failures, and other potential problems in your response.
7. **Repository:** Create a public repository on GitHub with your solution. Document your design choices, trade-offs, and provide clear instructions on how to run your application.
8. **Cloud Services:** If your solution requires any cloud services, please provide instructions on how to set them up. Please **do not** post them in your public repository.

Evaluation Criteria:

Your solution will be evaluated based on the following criteria:

1. Code Quality
2. Solution Design
3. Efficiency
4. Completeness
5. Documentation

6. Error Handling

Time Frame:

Please complete the test within 1 week and share the link to your GitHub repository with us.

Remember, the goal of this test is to showcase your skills and approach to building a data pipeline. Good luck!