# Peer Review

**Ankit Kumar's Approach:**

- Flask is being used to define various routes to handle different HTTP requests. The code uses pandas and pyspark libraries to manipulate data.
- The code has commented out route definitions that should be removed.
- The code can use some error handling when making requests to external APIs, for example, the `show_api_data()` function should have error handling for the case when the API is down or returns invalid data.
- Hide sensitive information, such as API keys, in environment variables instead of hard-coding them in the code.
- Code works good and the approach of developing a flask application for this use-case shows the results better.

**Shishir Singh's Approach:**

1. In the app.py file, a Flask application has been created and all the endpoints have been defined.
2. The required modules have been imported, and the developer has used df.createOrReplaceTempView('table') to execute SQL queries on a view.
3. All the endpoints are returning the results of the queries.
4. The code uses functions such as collect(), limit(), and desc().
5. Used spark.sql to execute the queries.
6. In the demo.py file, data has been cleaned and retrieved from the API.