Fatkhullakh Turakhonov (192066), Mark Slipenkyi (196668)

Database name: appointify_196668_192066

Project Report

The goal of this project was to create a data warehouse for managing appointment-related data. The solution fulfills the requirements by implementing:

1. Three types of storage formats: **Textfile**, **Parquet**, and **ORC**.
2. Internal and external tables.
3. Partitioning (both static and dynamic).
4. Use of complex data types such as STRUCT and ARRAY.
5. Bucketing is optional and not implemented due to the project's scope.

Key Tables and Design Choices

1. **Appointment Table**
    a. **Storage Type:** ORC (efficient for big tables and analytics).
    b. **Partitioning:** Partitioned by CompanyID.
    c. **Reasoning:** Appointment data is the biggest table and is frequently queried, making ORC an ideal format for compression and query performance.
2. **Company Table**
    a. **Storage Type:** External Table (Textfile).
    b. **Complex Types:** Includes STRUCT for Details (e.g., Type and Rating of the company).
    c. **Reasoning:** Company data may be shared or reused across systems.
3. **Customer Table**
    a. **Storage Type:** Textfile (External).

    b. **Partitioning:** By Gender for static partitioning.
    c. **Reasoning:** Useful for gender-specific queries without impacting performance.
4. **Date Table**
    a. **Storage Type:** Parquet.
    b. **Partitioning:** By Year for dynamic partitioning.
    c. **Reasoning:** Parquet optimizes storage and retrieval for big monotonic tables.
5. **Time Table**
    a. **Storage Type:** Parquet.
    b. **Partitioning:** By TimeOfDay.
    c. **Reasoning:** Same logic as Date.
6. **Worker Table**
    a. **Storage Type:** External Table (Textfile).
    b. **Complex Types:** Includes STRUCT for Details and WorkInfo.
    c. **Reasoning:** Worker data with nested fields allows detailed and structured querying.
7. **Service Type Table**
    a. **Storage Type:** External Table (Textfile).
    b. **Reasoning:** Holds static service information and requires no frequent updates.
8. **Junk Table**
    a. **Storage Type:** Textfile.
    b. **Reasoning:** Small table that handles miscellaneous data for cleanup and reference.

Competency Questions and Scenarios

1. **How does the appointment data differ based on the gender of the customers in terms of total appointments, revenue, discount utilization, and appointment duration?**

2. **Which workers handle the most appointments?**
    a. Queries Appointment_orc and Worker.
3. **How do appointment volumes vary on holidays versus working days?**
    a. Joins Appointment_orc with Date_par on isHoliday.
4. **What are the busiest days of the week?**
    a. Groups appointments by DayOfWeek.
5. **Which companies offer the highest discounts?**
    a. Aggregates data from Appointment_orc and Company.
6. **What are the appointment volumes for different service categories within each company?**
    a. Combines Company, ServiceType, and Appointment_orc.
7. **What is the revenue distribution by time of day?**
    a. Uses Time_par partitions and Appointment_orc.
8. **What is the average discount offered during holidays?**
    a. Aggregates discount data using isHoliday.
9. **What is the average cost of appointments based on worker ratings?**
    a. Joins Worker and Appointment_orc for grouped averages.
10. **What is the average duration of appointments by time of day?**
    a. Analyzes appointment durations using Time_par partitions.

Queries

The full HQL query scripts are attached, and each competency question corresponds to one query.

Explanation of Design Choices

1. **Storage Formats**
    a. ORC and Parquet were chosen for high-compression and fast analytics.
    b. Textfile was used for raw, external data.

2. **Partitioning**
    a. Dynamic and static partitioning reduced unnecessary data scans and improved query performance.
3. **Complex Types**
    a. Used STRUCT for worker and company details to simplify nested information access.
4. **External Tables**
    a. Enabled reusability and readibily for human eye.