CitySwift Data Operations and Performance Optimization

Project Overview

Objective: The purpose of this project is to monitor, clean, and analyze public transport data, including AVL (Automatic Vehicle Location), ticketing, and scheduling data, to identify inefficiencies and optimize performance. This documentation outlines the key queries and processes for ensuring data quality, analyzing key performance metrics, and generating actionable insights to enhance public transport operations.

Key Responsibilities:

- Ensure data quality through validation, cleaning, and optimization.
- Monitor and resolve technical issues related to data processing.
- Analyze performance metrics for routes, drivers, fuel consumption, and passenger demand.
- Generate reports and provide insights for product management and engineering teams.

Link to git: https://github.com/abhi921999/cityswift_job

Portfolio Link: https://abhi921999.github.io/My_Portfolio/

Table of Contents

1. Data Validation and Cleaning

Objective:

Ensure data integrity by identifying and addressing missing values and duplicates in the dataset, which is critical for reliable analysis and reporting.

Null Value Check:

Detect records with missing data to ensure that only complete and accurate data is used for analysis.

Duplicate Data:

Identify and remove duplicate records based on key fields to prevent data skewing.

```
-- To check if there are null values
SELECT * FROM [cityswift_insight].[dbo].[public_transport_data]
WHERE AVL_id IS NULL OR Ticket_id IS NULL  OR Bus_id IS NULL
OR Driver_id IS NULL OR Route_id IS NULL OR Timestamp IS NULL
OR Bus_Speed IS NULL OR Passenger_Count IS NULL OR Scheduled_Time IS NULL
OR Actual_Time IS NULL;

--checking duplicacy
DELETE FROM [cityswift_insight].[dbo].[public_transport_data]
WHERE AVL_id NOT IN (
    SELECT MIN(AVL_id)
    FROM public_transport_data
    GROUP BY Ticket_id, Schedule_id, Bus_id, Driver_id, Route_id, Timestamp
);
```

Outcome:
Data is clean and ready for analysis with no missing or duplicate records.

2. Key Performance Metrics Reporting

Objective:
Generate daily reports to monitor public transport operations, focusing on total trips, delays, fuel consumption, and passenger counts.

```
--kpis
SELECT CAST(Timestamp AS DATE) AS Date,
 COUNT(*) AS Total_Trips,  AVG(Route_Delay) AS Avg_Delay, AVG(Bus_Fuel_Consumption) AS Avg_Fuel_Consumption,  SUM(Passenger_Count) AS Total_Passenger_Count
FROM public_transport_data
GROUP BY CAST(Timestamp AS DATE)
ORDER BY Date DESC;
```

Outcome:
A daily report providing insights into total trips, delays, fuel consumption, and passenger counts, helping to track operational performance.

3. Route Performance Analysis

Objective:
Identify underperforming routes by analyzing average delay times, prioritizing those with the most significant delays for further investigation and optimization.

```
--top 10 routes with the most delays and their respective average delay times.
SELECT top 10 Route_id,
 AVG(Route_Delay) AS Average_Delay
FROM public_transport_data
GROUP BY Route_id
ORDER BY Average_Delay DESC
```

Outcome:
The top 10 routes with the most significant delays are identified for performance improvement initiatives.

4. Driver Performance Metrics

Objective:
Evaluate driver performance using metrics such as average delay, fuel consumption, and behavior scores. This helps identify top performers and areas where further training may be needed.

```
--Driver Performance Metrics
SELECT Driver_id,  AVG(Route_Delay) AS Avg_Delay,  AVG(Bus_Fuel_Consumption) AS Avg_Fuel_Consumption,
AVG(Bus_Behavior_Score) AS Avg_Behavior_Score
FROM public_transport_data
GROUP BY Driver_id
ORDER BY Avg_Behavior_Score DESC, Avg_Delay ASC, Avg_Fuel_Consumption ASC;
```

Outcome:
Drivers are ranked based on their performance metrics, enabling targeted interventions and recognition for top performers.

5. Passenger Demand Analysis

Objective:
Analyze passenger demand across different times of the day and days of the week to optimize bus schedules and ensure efficient resource allocation.

```sql
--Analyze Passenger Demand by Day of the Week
SELECT DATEPART(WEEKDAY, Timestamp) AS Day_of_Week, SUM(Passenger_Count) AS Total_Passenger_Count,
 AVG(Passenger_Count) AS Avg_Passenger_Count
FROM public_transport_data
GROUP BY DATEPART(WEEKDAY, Timestamp)
ORDER BY Total_Passenger_Count DESC;

--Analyze Passenger Demand by hour
SELECT DATEPART(HOUR, Timestamp) AS Hour_of_Day,  SUM(Passenger_Count) AS Total_Passenger_Count,
 AVG(Passenger_Count) AS Avg_Passenger_Count
FROM public_transport_data
GROUP BY DATEPART(HOUR, Timestamp)
ORDER BY Total_Passenger_Count DESC;
```

Outcome:
Demand patterns are identified, enabling optimized scheduling during peak and non-peak hours.

6. Fuel Efficiency Optimization

Objective:
Optimize fuel consumption by identifying routes and buses with high fuel usage, and reassigning resources to reduce operational costs.

```sql
--Fuel Efficiency Optimization
SELECT Route_id,  Bus_id, AVG(Bus_Fuel_Consumption) AS Avg_Fuel_Consumption
FROM public_transport_data
GROUP BY Route_id, Bus_id
ORDER BY Avg_Fuel_Consumption DESC;
```

Outcome:
Routes and buses with high fuel consumption are identified, allowing for cost-saving adjustments and optimized resource allocation.

7. Schedule Adherence

Objective:
Measure adherence to scheduled times, identify causes of deviation, and suggest corrective actions for improved schedule performance.

```sql
--Schedule Adherence
SELECT Route_id,  COUNT(*) AS Total_Trips, SUM(CASE WHEN Actual_Time <= Scheduled_Time THEN 1 ELSE 0 END) AS On_Time_Trips,
(SUM(CASE WHEN Actual_Time <= Scheduled_Time THEN 1 ELSE 0 END) * 100.0 / COUNT(*)) AS Adherence_Rate
FROM public_transport_data
GROUP BY Route_id
ORDER BY Adherence_Rate DESC;
```

Outcome:
A summary of schedule adherence rates by route is generated, highlighting areas needing adjustments to improve on-time performance.

8. Delay Analysis

Objective:

Conduct a comprehensive analysis of delays across the system, focusing on identifying the most delayed trips, classifying delays, and investigating underperforming routes.

```sql
--Query to retrieve bus trips where the route delay is greater than 30 minutes.
SELECT Bus_id, Route_id, Passenger_Count, Route_Delay
FROM public_transport_data
WHERE Route_Delay > 30
ORDER BY Route_Delay DESC;

-- Query to find routes with an average delay greater than 15 minutes
SELECT Route_id, AVG(Route_Delay) AS Avg_Delay
FROM public_transport_data
GROUP BY Route_id
HAVING AVG(Route_Delay) > 15
ORDER BY Avg_Delay DESC;

---- Query to retrieve trips for routes that have an average delay greater than the overall system average
SELECT Bus_id, Route_id, Route_Delay
FROM public_transport_data
WHERE Route_id IN ( SELECT Route_id FROM public_transport_data
    GROUP BY Route_id
    HAVING AVG(Route_Delay) > (
        SELECT AVG(Route_Delay)
        FROM public_transport_data
    )
)
ORDER BY Route_Delay DESC;

---- Query to find the trip with the highest delay for each route
WITH RankedTrips AS (
 SELECT Route_id, Bus_id, Route_Delay,
  ROW_NUMBER() OVER (PARTITION BY Route_id ORDER BY Route_Delay DESC) AS row_num
  FROM public_transport_data
)
SELECT Route_id, Bus_id, Route_Delay
FROM RankedTrips
WHERE row_num = 1;

---- Query to classify trips based on delay and count the number of trips in each category
SELECT CASE
       WHEN Route_Delay = 0 THEN 'On Time'
       WHEN Route_Delay BETWEEN 1 AND 15 THEN 'Slight Delay'
       WHEN Route_Delay BETWEEN 16 AND 30 THEN 'Moderate Delay'
       ELSE 'Severe Delay'
 END AS Delay_Category, COUNT(*) AS Trip_Count
FROM public_transport_data
GROUP BY  CASE
       WHEN Route_Delay = 0 THEN 'On Time'
       WHEN Route_Delay BETWEEN 1 AND 15 THEN 'Slight Delay'
       WHEN Route_Delay BETWEEN 16 AND 30 THEN 'Moderate Delay'
       ELSE 'Severe Delay'
 END;
```

Outcome:

The delay analysis identifies trips with significant delays, categorizes trips by delay severity, and helps pinpoint routes requiring further investigation.

9.Conclusion

- Several routes show consistent delays, necessitating further optimization.
- Passenger demand varies significantly by day and time, indicating a need for dynamic scheduling adjustments.
- Fuel consumption varies across routes and buses, presenting opportunities for cost savings through optimized resource allocation.
- Schedule adherence issues are prevalent on specific routes, requiring timetable adjustments and operational improvements.