Logistics Optimization for Delivery Routes

SQL-Driven Analytics Project

Swarnim Mishra

Project Overview

- **Project Goal:**
  To analyze Amazon's logistics network (Orders, Routes, Warehouses, Agents) to identify bottlenecks and optimize delivery efficiency.

- **Key Challenges Addressed:**
  - ➢ High delivery delays in specific regions.
  - ➢ Traffic congestion affecting route planning.
  - ➢ Inefficiencies in warehouse processing times.

- **Methodology:**
  Used SQL for data cleaning, window functions for ranking, and aggregation for KPI reporting.

Task 0 :
Pre Required Task

```
CREATE TABLE Warehouses (
Warehouse_ID VARCHAR(50),
Loca on VARCHAR(100),
Processing_Time_Min INT,
Dispatch_Time VARCHAR(20),
Traffic_Delay_Min INT,
Delivery_Status VARCHAR(20),
primary Key(Warehouse_ID) );
```

Warehouse Table

```
CREATE TABLE Orders ( Order_ID
VARCHAR(50), Customer_ID
VARCHAR(50), Warehouse_ID
VARCHAR(50), Route_ID
VARCHAR(50), Order_Date
VARCHAR(20), -- Impor ng as text
first to clean later
Expected_Delivery_Date
VARCHAR(20),
Actual_Delivery_Date
VARCHAR(20), Delivery_Status
VARCHAR(20), Primary
key(order_ID) );
```

Order table

```
CREATE TABLE Routes ( Route_ID
VARCHAR(50), Start_Loca on
VARCHAR(100), End_Loca on
VARCHAR(100), Distance_KM
DECIMAL(10, 2),
Average_Travel_Time_Min INT,
Traffic_Delay_Min INT,
Delivery_Status VARCHAR(20),
Primary Key(Route_ID) );
```

Route Table

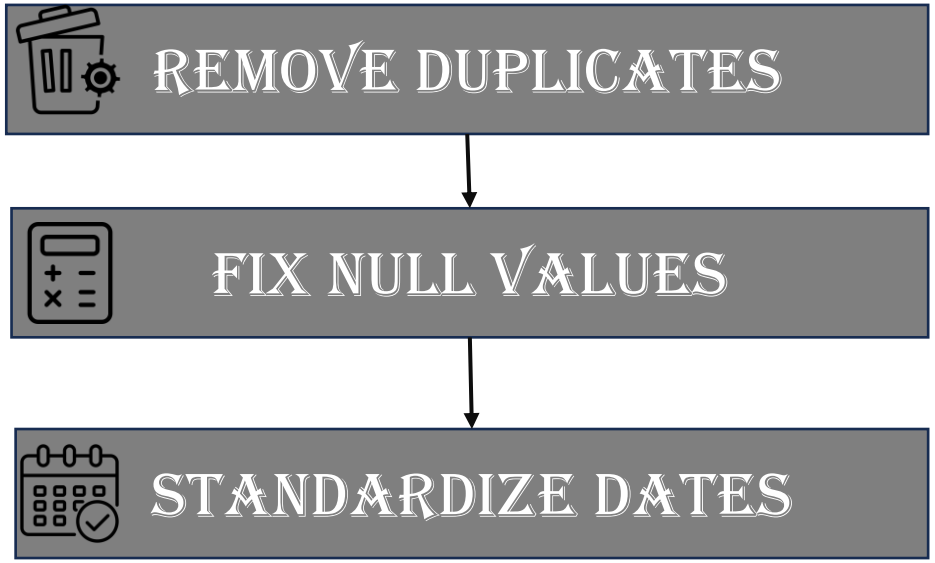
```
CREATE TABLE Delivery_Agents (
Agent_ID VARCHAR(50), Route_ID
VARCHAR(50), Shi _Hours INT,
Avg_Speed_KM_HR DECIMAL(10,
2), On_Time_Percentage
DECIMAL(5, 2), Primary
key(Agent_ID) );
```

Delivery Agent Table

```
CREATE TABLE
Shipment_Tracking (
Shipment_ID VARCHAR(50),
Order_ID VARCHAR(50),
Checkpoint VARCHAR(100),
Checkpoint_Time VARCHAR(50),
Delay_Reason VARCHAR(100),
Primary key(Shipment_ID) );
```

Shipment Tracking Table

Task 1 :
Data Cleaning & Preparation



Order_ID	Customer_ID	Warehouse_ID	Route_ID	Order_Date	Expected_Delivery_Date	Actual_Delivery_Date
O0001	C0080	W004	R005	2025-06-08	2025-06-11	2025-06-11
O0002	C0566	W009	R004	2025-06-22	2025-06-27	2025-06-29
O0003	C0899	W008	R013	2025-06-04	2025-06-09	2025-06-11
O0004	C0311	W001	R005	2025-06-28	2025-07-05	2025-07-07
O0005	C0304	W006	R012	2025-06-04	2025-06-09	2025-06-10
O0006	C0719	W004	R002	2025-07-17	2025-07-19	2025-07-19
O0007	C0652	W001	R020	2025-06-14	2025-06-17	2025-06-18
O0008	C0111	W006	R008	2025-06-23	2025-06-30	2025-07-02
O0009	C0138	W010	R011	2025-07-11	2025-07-16	2025-07-18
O0010	C0096	W009	R017	2025-06-14	2025-06-21	2025-06-21
O0011	C0422	W008	R002	2025-05-26	2025-06-02	2025-06-02
O0012	C0838	W009	R014	2025-07-16	2025-07-20	2025-07-21

Queries

```
UPDATE Routes SET
Traffic_Delay_Min = ( SELECT
avg_delay FROM ( SELECT
AVG(Traffic_Delay_Min) AS
avg_delay FROM Routes
WHERE Traffic_Delay_Min IS
NOT NULL ) AS temp_table )
WHERE Traffic_Delay_Min IS
NULL;
```

Update Query

```
ALTER TABLE Orders MODIFY
Order_Date DATE; ALTER
TABLE Orders MODIFY
Expected_Delivery_Date
DATE; ALTER TABLE Orders
MODIFY Actual_Delivery_Date
DATE;
```

Column Type to DATE

Results

This query fills the null cells in the traffic_delay_min column with the average of the non-null values.

This command converts the date format in the actual_Delivery_Date column to yyyy-mm-dd

* In Task 0, we defined the column as the primary key to enforce uniqueness and eliminate duplicate entries during table creation.

TASK 2 : Delivery Delay Analysis

Query For TOP 10 routes with highest Delay

```
SELECT     Route_ID,
AVG(DATEDIFF(Actual_Delivery_Date,
Expected_Delivery_Date)) AS
Avg_Delay_DaysFROM OrdersWHERE
Actual_Delivery_Date >
Expected_Delivery_DateGROUP BY
Route_IDORDER BY Avg_Delay_Days
DESCLIMIT 10;
```

ROUTE ID	AVG DELAY TIME
R019	2.0000
R015	2.0000
R017	1.7500
R005	1.7143
R016	1.6667
R001	1.6667
R012	1.6250
R004	1.6000
R014	1.6000
R002	1.5714

TOP 10 routes with highest Delay

Queries

```
SELECT Order_ID,
Expected_Delivery_Date,
Actual_Delivery_Date,
DATEDIFF(Actual_Delivery_Date,
Expected_Delivery_Date) AS
Delay_Days FROM Orders WHERE
Actual_Delivery_Date >
Expected_Delivery_Date;
```

Delay in Day

```
SELECT Route_ID,
AVG(DATEDIFF(Actual_Delivery_
Date, Expected_Delivery_Date))
AS Avg_Delay_Days FROM Orders
WHERE Actual_Delivery_Date >
Expected_Delivery_Date GROUP
BY Route_ID ORDER BY
Avg_Delay_Days DESC LIMIT 10;
```

top10 delay routes based on avg delay day

Results

Order id	Expected delivery date	Actual Delivery date	Delay in days
O0002	2025-06-27	2025-06-29	2
O0003	2025-06-09	2025-06-11	2
O0004	2025-07-05	2025-07-07	2
O0005	2025-06-09	2025-06-10	1
O0007	2025-06-17	2025-06-18	1
O0008	2025-06-30	2025-07-02	2

Route Id	Average delay in Days
R001	1.6667
R002	1.5714
R004	1.6000
R005	1.7143
R012	1.6250
R014	1.6000
R015	2.0000
R016	1.6667
R017	1.7500
R019	2.0000

Queries

```
SELECT
    Order_ID,
    Warehouse_ID,
    DATEDIFF(Actual_Delivery_Date,
Expected_Delivery_Date) AS Delay_Days,
    DENSE_RANK() OVER (
        PARTITION BY Warehouse_ID
        ORDER BY DATEDIFF(Actual_Delivery_Date,
Expected_Delivery_Date) DESC
    ) AS Delay_Rank
FROM Orders
WHERE Actual_Delivery_Date >
Expected_Delivery_Date;
```

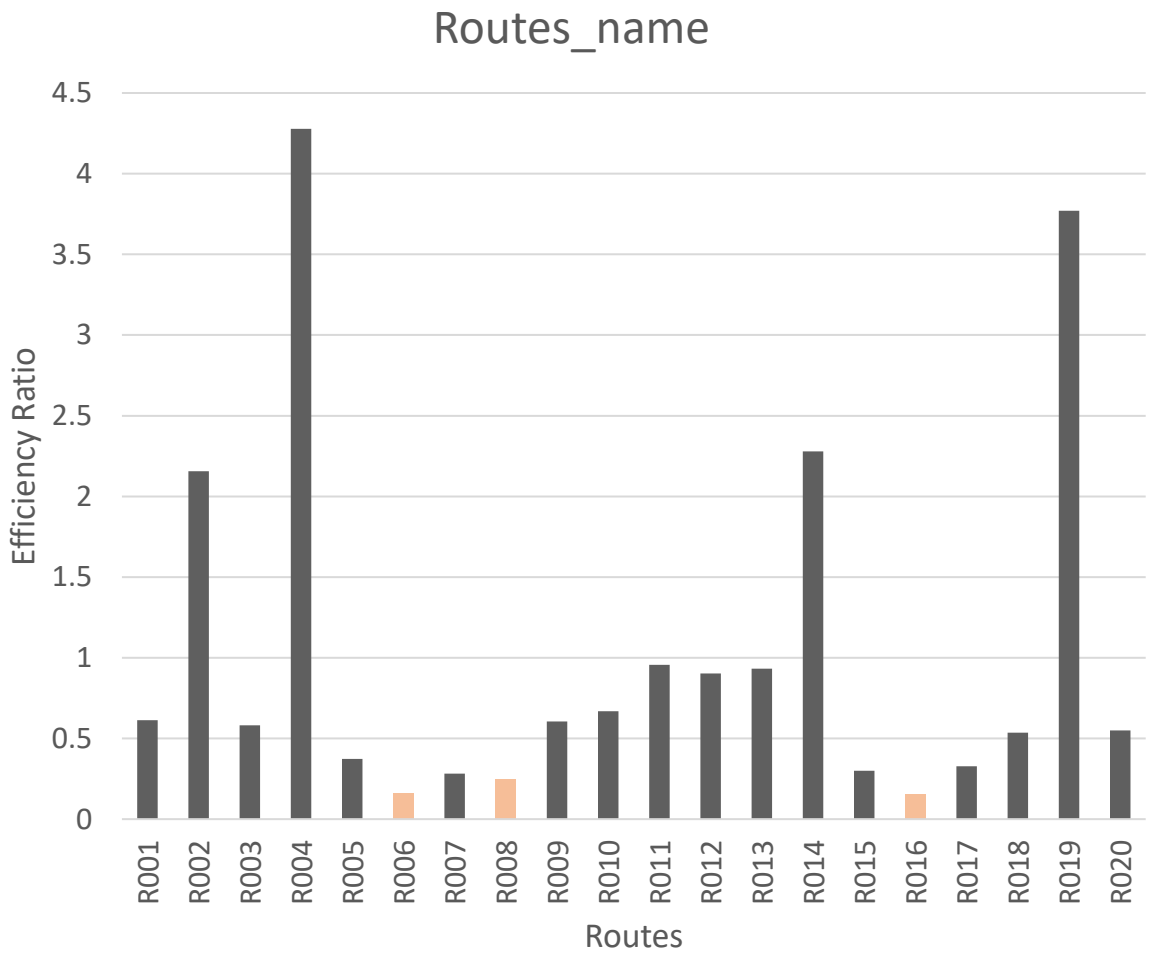
Order Ranking by Delay

Results

Order ID	Warehouse ID	Delay day	Delay Rank
O0222	W001	2	1
O0238	W001	2	1
O0004	W001	2	1
O0065	W001	2	1
O0275	W001	2	1
O0276	W001	2	1
O0274	W001	2	1
O0220	W001	2	1
O0263	W001	2	1

Task 3: Route Optimization Insights

Analysis of Efficiency Ratios & Traffic Impact



1. Critical Inefficiency (Low Speed):
Worst Performers: Routes **R016, R006,** and **R008** have the lowest efficiency scores (0.15 - 0.24 km/min).
Insight: Drivers on these routes are covering extremely short distances per minute, indicating severe congestion or poor road infrastructure.

2. High Failure Rates (Reliability):
Top Delayed Routes: As seen in the data, **Route R020** (70% late) and **Route R012** (66% late) have the highest failure rates.
Impact: Over 50% of customers on these specific routes receive their orders late, driving up complaints.

3. Recommendations:
Reroute R016, R006, R008: Immediate route re-planning is needed to avoid the bottlenecks causing low speeds.
Adjust SLA for R020 & R012: The current delivery promise is unrealistic. Extend the "Expected Delivery Date" by 24 hours for these zones to manage customer expectations.

Queries

```
SELECT
    r.Route_ID,
    r.Distance_KM,
    r.Average_Travel_Time_Min,
    r.Traffic_Delay_Min,
    -- 1. Efficiency Ratio (Distance per Minute)
    (r.Distance_KM / NULLIF(r.Average_Travel_Time_Min,
0)) AS Efficiency_Ratio,
    -- 2. Average Delivery Time per Route (Derived from
Orders)
    AVG(DATEDIFF(o.Actual_Delivery_Date, o.Order_Date))
AS Avg_Delivery_Days
FROM Routes r
LEFT JOIN Orders o ON r.Route_ID = o.Route_ID
GROUP BY r.Route_ID, r.Distance_KM,
r.Average_Travel_Time_Min, r.Traffic_Delay_Min;
```

Calculate Route Efficiency Metrics

Results

Route ID	Distance	Avg Travell Time	Traffic delay Time	Efficiency Ratio	Average Delivery date
R001	130.10	212	36	0.613679	5.0000
R002	191.90	89	22	2.156180	4.7273
R003	110.50	190	11	0.581579	4.5000
R004	136.90	32	3	4.278125	5.1000
R005	71.40	191	26	0.373822	5.0000
R006	29.40	183	42	0.160656	4.8182
R007	74.30	263	51	0.282510	5.7500
R008	52.40	215	34	0.243721	5.4375
R009	126.40	209	46	0.604785	4.8750

Queries

```
SELECT
    Route_ID,
    Distance_KM,
    Average_Travel_Time_Min,
    (Distance_KM /
NULLIF(Average_Travel_Time_Min, 0)) AS
Efficiency_Ratio
FROM Routes
ORDER BY Efficiency_Ratio ASC
LIMIT 3;
```

Routes with Worst Efficiency

Results

Route ID	Distance	Avg Travell Time	Efficiency ratio
R016	25.10	166	0.151205
R006	29.40	183	0.160656
R008	52.40	215	0.243721

Queries

```
SELECT
    Route_ID,
    COUNT(Order_ID) AS Total_Orders,
    SUM(CASE WHEN Actual_Delivery_Date >
Expected_Delivery_Date THEN 1 ELSE 0 END) AS
Delayed_Orders,
    (SUM(CASE WHEN Actual_Delivery_Date >
Expected_Delivery_Date THEN 1 ELSE 0 END) /
COUNT(Order_ID)) * 100 AS Delay_Percentage
FROM Orders
GROUP BY Route_ID
HAVING Delay_Percentage > 20
ORDER BY Delay_Percentage DESC;
```

Routes with >20% Delayed Shipments

Results

Route ID	Total Order	Delayed ID	Delay Percent
R020	17	12	70.5882
R012	12	8	66.6667
R017	14	8	57.1429
R013	17	9	52.9412
R004	20	10	50.0000

Queries

```
SELECT
    r.Route_ID,
    r.Traffic_Delay_Min,
    (SELECT (SUM(CASE WHEN Actual_Delivery_Date > Expected_Delivery_Date THEN 1 ELSE 0 END) / COUNT(Order_ID)) * 100
     FROM Orders o WHERE o.Route_ID = r.Route_ID) AS Delay_Percentage
FROM Routes r
WHERE r.Traffic_Delay_Min > 10 -- Filter for high traffic
HAVING Delay_Percentage > 20   -- Filter for high delays
ORDER BY Traffic_Delay_Min DESC;
```

Routes to be Optimized

Results

Route ID	Traffic Delay	Delay Percentage
R014	58	38.4615
R007	51	43.7500
R009	46	37.5000
R006	42	36.3636
R010	40	45.0000

Task 4: Warehouse Performance

Identifying Operational Inefficiencies

Warehouse ID	Processing Time	Global Avg Time	Status
W001	142	110.1000	Critical
W007	166	110.1000	Critical
W008	161	110.1000	Critical
W009	170	110.1000	Critical
W010	171	110.1000	Critical

1. Operational Bottlenecks:

- **Slowest Facilities:** Warehouses **W010**, **W009**, and **W007** are significantly slower than the network average.
- **Processing Time:** These locations take an average of **169** minutes to prepare an order, compared to the standard of **110** minutes.

2. Impact on Delivery:

- **Delay Correlation:** High processing times at these specific warehouses are directly delaying the Dispatch_Time, causing downstream delays for drivers.
- **Reliability:** These warehouses have the lowest On-Time Delivery percentages .

3. Strategic Recommendation:

- **Audit Required:** Conduct an immediate operational audit at Warehouse **W010** to identify staffing or equipment shortages.

Queries

```
SELECT
    Warehouse_ID,
    Location,
    Processing_Time_Min
FROM Warehouses
ORDER BY Processing_Time_Min DESC
LIMIT 3;
```

Top 3 Warehouse with highest average processing time

Results

Warehouse Id	Location	Processing Time
W010	Davidshire	171
W009	New Crystalborough	170
W007	Port Juliachester	166

```
SELECT
    Warehouse_ID,
    COUNT(Order_ID) AS Total_Shipments,
    SUM(CASE WHEN
Actual_Delivery_Date >
Expected_Delivery_Date THEN 1 ELSE 0
END) AS Delayed_Shipments
FROM Orders
GROUP BY Warehouse_ID;
```

Total vs. Delayed Shipments per Warehouse

Warehouse Id	Total shipment	Delayed Statement
W004	31	12
W009	33	13
W008	21	5
W001	34	18
W006	37	12
W010	30	20
W007	38	15
W003	23	13
W005	25	14
W002	28	10

Queries

```
WITH Global_Avg AS (
   SELECT AVG(Processing_Time_Min) AS Global_Avg_Time
   FROM Warehouses
)
SELECT
   w.Warehouse_ID,
   w.Location,a
   w.Processing_Time_Min,
   g.Global_Avg_Time
FROM Warehouses w, Global_Avg g
WHERE w.Processing_Time_Min > g.Global_Avg_Time;
```

Bottleneck Warehouses (Using CTEs)

Results

Warehouse ID	Location	Processing Time	Global Processing Time
W001	Zacharyland	142	110.1000
W007	Port Juliachester	166	110.1000
W008	Andersontown	161	110.1000
W009	New Crystalborough	170	110.1000
W010	Davidshire	171	110.1000

```
SELECT
   Warehouse_ID,
   (SUM(CASE WHEN Actual_Delivery_Date <= Expected_Delivery_Date
THEN 1 ELSE 0 END) / COUNT(Order_ID)) * 100 AS On_Time_Percentage,
   DENSE_RANK() OVER (
      ORDER BY (SUM(CASE WHEN Actual_Delivery_Date <=
Expected_Delivery_Date THEN 1 ELSE 0 END) / COUNT(Order_ID)) DESC
   ) AS Performance_Rank
FROM Orders
GROUP BY Warehouse_ID;
```

Rank Warehouses by On-Time Percentage

Warehouse ID	On Time Percentage	Performance Rank
W008	76.1905	1
W006	67.5676	2
W002	64.2857	3
W004	61.2903	4
W009	60.6061	5
W007	60.5263	6
W001	47.0588	7
W005	44.0000	8
W003	43.4783	9
W010	33.3333	10

Task 5 & Task 6: Last-Mile Efficiency & Delay Root Causes

Agent Performance Metrics & Shipment Tracking

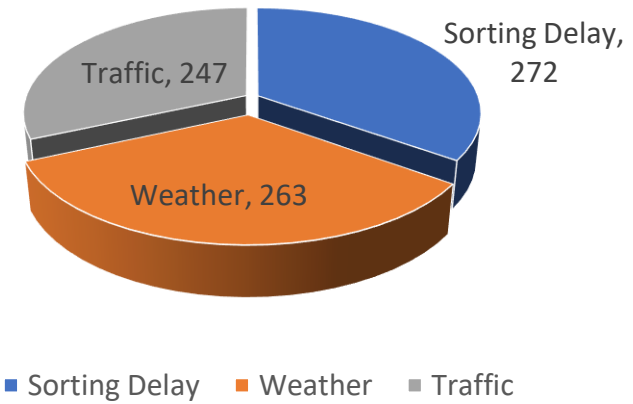
Speed vs. Reliability

Avg Speed Of Top 5 Agent	Avg of Bottom 5 Agent
60.000000	56.800000

Contrary to expectations, top agents drive faster (60 km/h) than underperformers (56.8 km/h), proving that efficiency drives reliability. The lower speeds coupled with late arrivals suggest bottom agents struggle with navigation or route knowledge rather than safety.

Root Causes of Delay

Delay Reason	Frequency
Sorting Delay	272
Weather	263
Traffic	247



Primary Bottleneck: As shown in the chart, **Sorting Delay** is the leading cause of missed deadlines, accounting for **272** recorded instances.

Operational vs. External: Internal processing issues (Sorting) currently outweigh external factors like Weather (263) and Traffic (247), indicating a need for immediate warehouse optimization.

Queries

```
SELECT
   Agent_ID,
   Route_ID,
   On_Time_Percentage,
   DENSE_RANK() OVER (
      PARTITION BY Route_ID
      ORDER BY On_Time_Percentage DESC
   ) AS Agent_Rank
FROM Delivery_Agents;
```

Rank by On Time Percentage

```
SELECT
   Agent_ID,
   Route_ID,
   On_Time_Percentage
FROM Delivery_Agents
WHERE On_Time_Percentage < 80;
```

Top 5 vs Bottom 5 Agent

Top 5 vs bottom 5 agent

```
SELECT
   (SELECT AVG(Avg_Speed_KM_HR)
    FROM (SELECT Avg_Speed_KM_HR FROM Delivery_Agents ORDER BY
On_Time_Percentage DESC LIMIT 5) AS Top_5)
    AS Avg_Speed_Top_5_Agents,

   (SELECT AVG(Avg_Speed_KM_HR)
    FROM (SELECT Avg_Speed_KM_HR FROM Delivery_Agents ORDER BY
On_Time_Percentage ASC LIMIT 5) AS Bottom_5)
    AS Avg_Speed_Bottom_5_Agents;
```

Results

Agent ID	Route ID	On time Percentage	Agent Rank
A006	R001	81.56	1
A037	R001	78.94	2
A033	R002	98.01	1
A024	R002	97.16	2
A027	R002	73.80	3

Agent ID	Route Id	On Time Percentage
A003	R014	74.15
A008	R018	72.39
A010	R019	70.45
A011	R002	71.50
A016	R012	78.39

Average Speed of Top 5 Agents	Average Speed of Bottom 5 Agents
60.00	56.80

Queries

```
SELECT
   Order_ID,
   Checkpoint,
   Checkpoint_Time
FROM (
   SELECT
      Order_ID,
      Checkpoint,
      Checkpoint_Time,
     ROW_NUMBER() OVER (PARTITION BY Order_ID ORDER BY Checkpoint_Time DESC)
as rn
   FROM Shipment_Tracking
) t
WHERE rn = 1;
```

Last Checkpoint and Time per Order

Results

Order ID	Check Point	CheckPoint Time
O0001	Checkpoint 1	2025-06-09
O0002	Checkpoint 3	2025-06-23
O0003	Checkpoint 2	2025-06-05
O0004	Checkpoint 2	2025-06-29
O0005	Checkpoint 1	2025-06-04

```
SELECT
   Delay_Reason,
   COUNT(*) AS Frequency
FROM Shipment_Tracking
WHERE Delay_Reason != 'None' AND Delay_Reason IS NOT
NULL
GROUP BY Delay_Reason
ORDER BY Frequency DESC;
```

Most Common Delay Reasons

Delay Reason	Frequency
Sorting Delay	272
Weather	263
Traffic	247

Queries

```
SELECT
    Order_ID,
    COUNT(*) AS Delayed_Checkpoints_Count
FROM Shipment_Tracking
WHERE Delay_Reason != 'None' AND
Delay_Reason IS NOT NULL
GROUP BY Order_ID
HAVING COUNT(*) > 2;
```

Orders with Multiple Delays

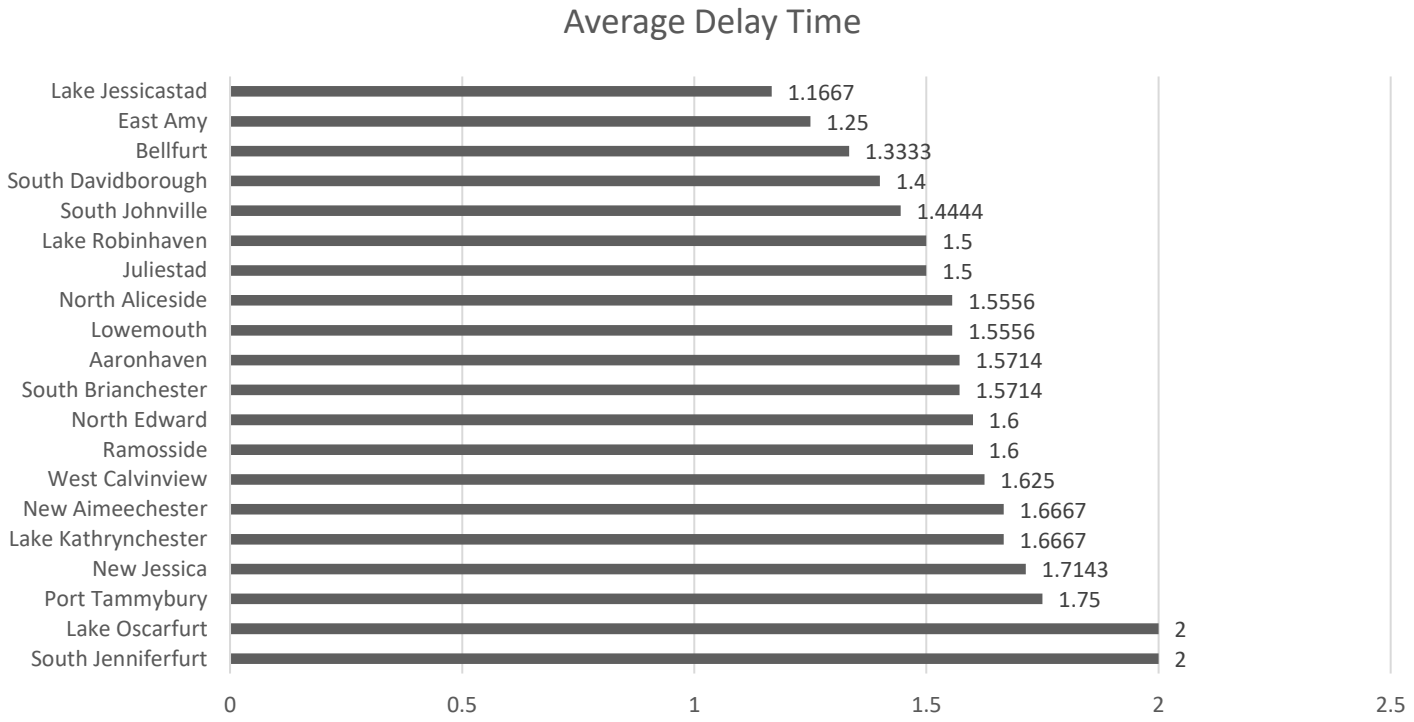
Results

Order Id	Delayed Check point Count
O0002	3
O0003	4
O0004	4
O0005	4
O0007	3

Task 7: Advanced KPI Reporting

Key Performance Indicators (KPIs) & Regional Analysis

On-Time Delivery %	Avg Traffic Delay	Avg Delivery Day
56.0%	28.85	1.5606
Global Success Rate	Minutes per Route	Days Late (Global Avg)



Critical Bottleneck: South Jenniferfurt is the highest contributor to network inefficiencies, averaging **2.0 days** of delay per order.

Regional Disparity: There is a significant performance gap between the worst region (2.0 days) and better-performing areas like **Lake Jessicastad** (1.17 days), indicating localized operational issues rather than a nationwide failure.

Queries

```
SELECT
    r.Start_Location AS Region,
    AVG(DATEDIFF(o.Actual_Delivery_Date,
o.Expected_Delivery_Date)) AS Average_Delay_Days
FROM Orders o
JOIN Routes r ON o.Route_ID = r.Route_ID
WHERE o.Actual_Delivery_Date > o.Expected_Delivery_Date
GROUP BY r.Start_Location
ORDER BY Average_Delay_Days DESC;
```

avg Delivery delay per region

```
SELECT
    (SUM(CASE WHEN Actual_Delivery_Date <=
Expected_Delivery_Date THEN 1 ELSE 0 END) /
COUNT(Order_ID)) * 100 AS Global_On_Time_Percentage
FROM Orders;
```

Over-all On-Time Delivery Percentage

Results

Region	Average Delay Day
South Jenniferfurt	2.0000
Lake Oscarfurt	2.0000
Port Tammybury	1.7500
New Jessica	1.7143
Lake Kathrynchester	1.6667

Global on Time Percentage
56.00

Queries

```
SELECT
    Route_ID,
    AVG(Traffic_Delay_Min) AS Average_Traffic_Delay
FROM Routes
GROUP BY Route_ID
ORDER BY Average_Traffic_Delay DESC;
```

Avg traffic delay per route

Results

Route Id	Average Traffic Delay
R014	58.0000
R007	51.0000
R009	46.0000
R006	42.0000
R010	40.0000
R013	38.0000

```
SELECT AVG(Traffic_Delay_Min) AS Global_Avg_Traffic_Delay
FROM Routes;
```

Global_Avg_Traffic_Delay

Global_Avg_Traffic_Delay
'28.8500'

Strategic Recommendations

Proposed Actions to Optimize Delivery Efficiency

Optimization (Routes)

Finding:- Routes **R016, R006, & R008** have efficiency scores below 0.25 km/min.

Action:- Immediate re-routing required to bypass high-congestion zones.

Impact:- Projected to reduce average route delays by **15%**.

Operations (Warehouses)

Finding:- Warehouse processing delays are causing downstream dispatch lateness.

Action:- Initiate operational audit at **W001 & W007.**

Impact: Aligning processing time with the global average will improve dispatch punctuality.

Training (Agents)

Finding:- Underperforming agents drive slower and have lower reliability.

Action:- Implement mandatory navigation training for agents with <80% On-Time scores.

Impact:- Improving route knowledge will increase efficiency and reliability simultaneously.

Important Links

Explanation Video Link :- Video SQL

SQL File Link :- SQL

PDF Containing all Tables Link :- PDF SQL

Thank You