TTC MACRO LEVEL ASSESSMENT

About TTC:

Toronto Transit is the public transport agency that operates bus, subway, streetcar, and paratransit services in Toronto, Ontario. There are around 2000 buses, 204 streetcars and 800 subway cars operating currently across the system. It facilitates over 1.6 million daily rides and is vital for day-to-day commute for a large number of citizens of Toronto. Yet, this extensive transport system does run into problems sometimes, and we hear a lot of incidents of delays and breakdowns in TTC.

In this report, we have tried to analyze the situation of TTC delays. We have tried to understand the factors contributing to delays and disruptions in TTC services by querying the data at hand and presenting it through visualizations, so that it can be used by the TTC management to identify the loopholes in the system and improve the situation.

Our dataset:

Our dataset consists of TTC delay data for two years: 2022 and 2023. We obtained our dataset from Kaggle

The dataset characteristics are: Date, Route, Time, Day, Location, Incident, Minimum Delay, Minimum Gap, and Direction in both the dataset.

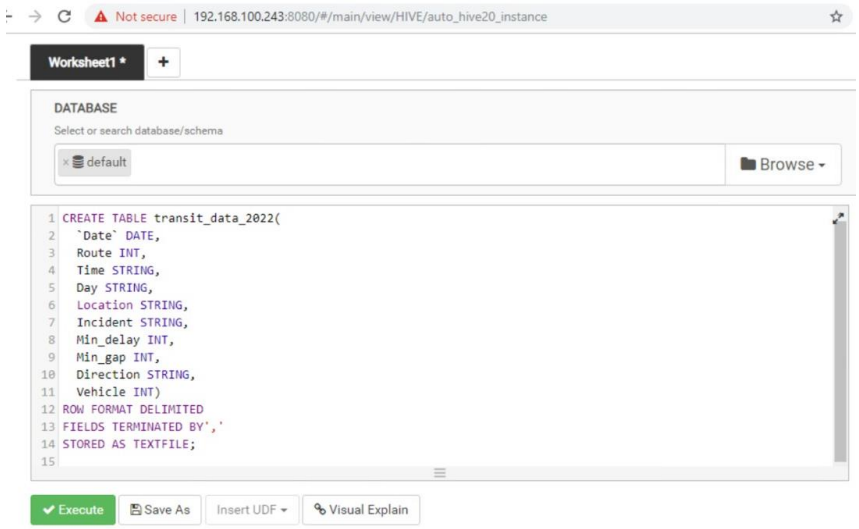
We aggregated dataset and made an aggregate dataset named *transit_data_combined* using **HIVE**. The dataset characteristics remain the same.

Our Tech stack:

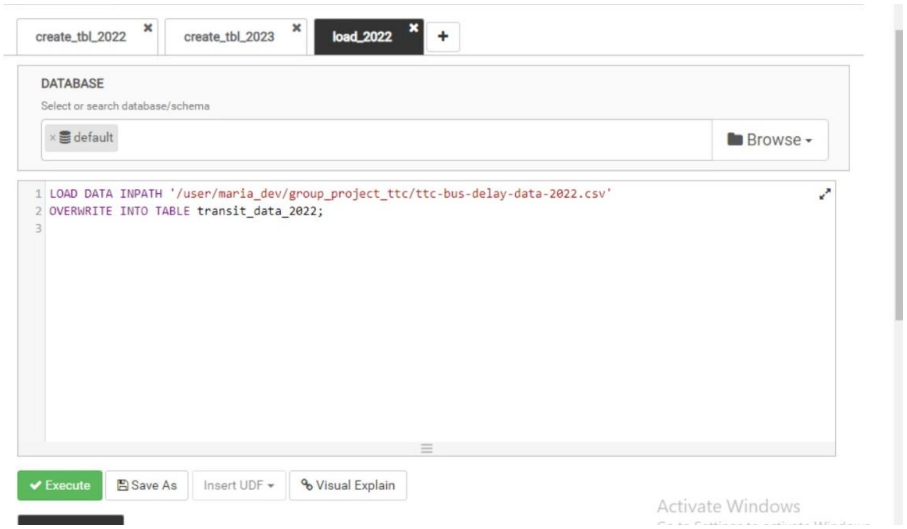
Platform	Use case
HIVE	Data Collection and Storage: To establish a centralized repository for TTC delay data. Data Processing: To create tables for each dataset and loaded data into them. Data Integration: To tables from 2022 and 2023 into a single table for comprehensive analysis.
ZEPPELIN	Data Visualization: To derive visual insights and analyze the combined data.

Creating, loading and combining datasets

1) CREATED TABLE TRANSIT_DATA _2022 TO LOAD THE DATA FOR THE YEAR 2022



2) LOADED THE DATA FOR 2022 IN THE ABOVE TABLE



3) CREATED TABLE TRANSIT_DATA _2023 TO LOAD THE DATA FOR THE YEAR 2023

```
CREATE TABLE transit_data_2023(
  `Date` DATE,
  Route INT,
  Time STRING,
  Day STRING,
  Location STRING,
  Incident STRING,
  Min_delay INT,
  Min_gap INT,
  Direction STRING,
  Vehicle INT)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY','
STORED AS TEXTFILE;
```

Execute | Save As | Insert UDF | Visual Explain

4) LOADED THE DATA FOR 2023 IN THE ABOVE TABLE

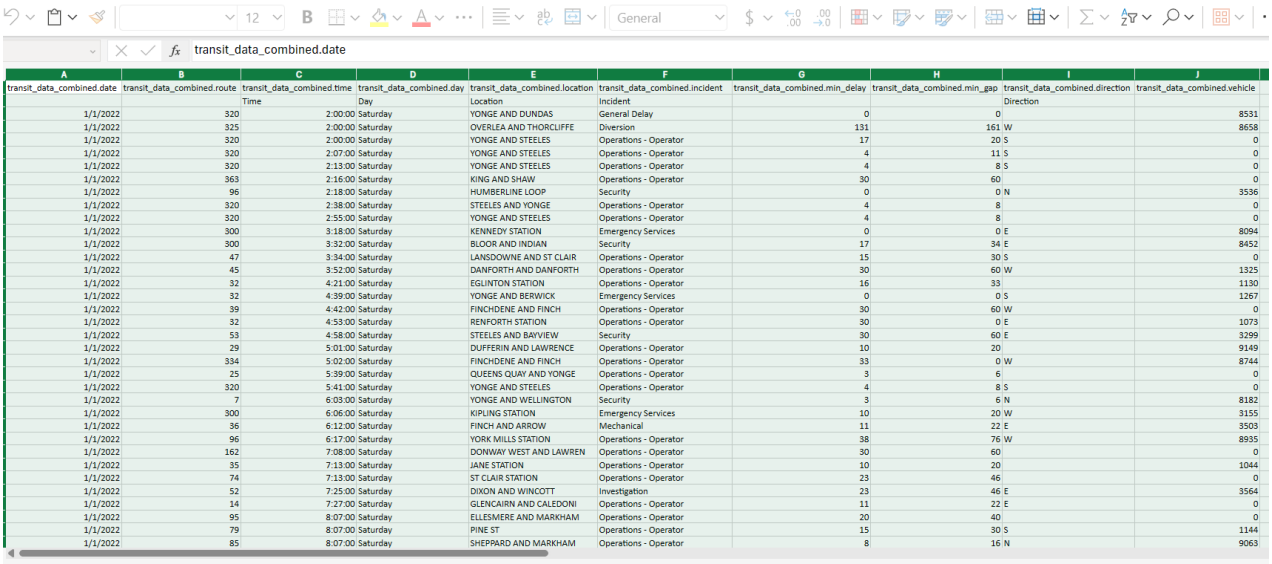
```
LOAD DATA INPATH '/user/maria_dev/group_project_ttc/ttc-bus-delay-data-2023.csv'
OVERWRITE INTO TABLE transit_data_2023;
```

Execute | Save As | Insert UDF | Visual Explain

5) CREATED TABLE TRANSIT_DATA_COMBINED TO MERGE BOTH THE TABLES

```
CREATE TABLE transit_data_combined(
  `Date` DATE,
  Route INT,
  Time STRING,
  Day STRING,
  Location STRING,
  Incident STRING,
  Min_delay INT,
  Min_gap INT,
  Direction STRING,
  Vehicle INT)
```

Execute | Save As | Insert UDF | Visual Explain



transit_data_combined.date	transit_data_combined.route	transit_data_combined.time	transit_data_combined.day	transit_data_combined.location	transit_data_combined.incident	transit_data_combined.min_delay	transit_data_combined.min_gap	transit_data_combined.direction	transit_data_combined.vehicle
		Time	Day	Location	Incident			Direction	
1/1/2022	320	2:00:00	Saturday	YONGE AND DUNDAS	General Delay	0	0		8531
1/1/2022	325	2:00:00	Saturday	OVERLEA AND THORCLIFFE	Diversion	131	161	W	8658
1/1/2022	320	2:00:00	Saturday	YONGE AND STEELES	Operations - Operator	17	20	S	0
1/1/2022	320	2:07:00	Saturday	YONGE AND STEELES	Operations - Operator	4	11	S	0
1/1/2022	320	2:13:00	Saturday	YONGE AND STEELES	Operations - Operator	4	8	S	0
1/1/2022	363	2:16:00	Saturday	KING AND SHAW	Operations - Operator	30	60		0
1/1/2022	96	2:18:00	Saturday	HUMBERLINE LOOP	Security	0	0	N	3536
1/1/2022	320	2:38:00	Saturday	STEELES AND YONGE	Operations - Operator	4	8		0
1/1/2022	320	2:55:00	Saturday	YONGE AND STEELES	Operations - Operator	4	8		0
1/1/2022	300	3:18:00	Saturday	KENNEDY STATION	Emergency Services	0	0	E	8094
1/1/2022	300	3:32:00	Saturday	BLOOR AND INDIAN	Security	17	34	E	8452
1/1/2022	47	3:34:00	Saturday	LANSDOWNE AND ST CLAIR	Operations - Operator	15	30	S	0
1/1/2022	45	3:52:00	Saturday	DANFORTH AND DANFORTH	Operations - Operator	30	60	W	1325
1/1/2022	32	4:21:00	Saturday	EGLINTON STATION	Operations - Operator	16	33		1130
1/1/2022	32	4:39:00	Saturday	YONGE AND BERWICK	Emergency Services	0	0	S	1267
1/1/2022	39	4:42:00	Saturday	FINCHDENE AND FINCH	Operations - Operator	30	60	W	0
1/1/2022	32	4:53:00	Saturday	RENFORTH STATION	Operations - Operator	30	0	E	1073
1/1/2022	53	4:58:00	Saturday	STEELES AND BAYVIEW	Security	30	60	E	3299
1/1/2022	29	5:01:00	Saturday	DUFFERIN AND LAWRENCE	Operations - Operator	10	20		9149
1/1/2022	334	5:02:00	Saturday	FINCHDENE AND FINCH	Operations - Operator	33	0	W	8744
1/1/2022	25	5:39:00	Saturday	QUEENS QUAY AND YONGE	Operations - Operator	3	6		0
1/1/2022	320	5:41:00	Saturday	YONGE AND STEELES	Operations - Operator	4	8	S	0
1/1/2022	7	6:03:00	Saturday	YONGE AND WELLINGTON	Security	3	6	N	8182
1/1/2022	300	6:06:00	Saturday	KIPLING STATION	Emergency Services	10	20	W	3155
1/1/2022	36	6:12:00	Saturday	FINCH AND ARROW	Mechanical	11	22	E	3503
1/1/2022	96	6:17:00	Saturday	YORK MILLS STATION	Operations - Operator	38	76	W	8935
1/1/2022	162	7:08:00	Saturday	DONWAY WEST AND LAWREN	Operations - Operator	30	60		0
1/1/2022	35	7:13:00	Saturday	JANE STATION	Operations - Operator	10	20		1044
1/1/2022	74	7:13:00	Saturday	ST CLAIR STATION	Operations - Operator	23	46		0
1/1/2022	52	7:25:00	Saturday	DIXON AND WINCOTT	Investigation	23	46	E	3564
1/1/2022	14	7:27:00	Saturday	GLENCAIRN AND CALEDONI	Operations - Operator	11	22	E	0
1/1/2022	95	8:07:00	Saturday	ELLESMERE AND MARKHAM	Operations - Operator	20	40		0
1/1/2022	79	8:07:00	Saturday	PINE ST	Operations - Operator	15	30	S	1144
1/1/2022	85	8:07:00	Saturday	SHEPPARD AND MARKHAM	Operations - Operator	8	16	N	9063

Visualization steps

Steps:

1. We created a new notebook in Zeppelin and loaded data into the data data frame.

> Val transit_data_combined = (spark.read.option("header","true").option("inferSchema","true").csv("/user/maria_dev/group_project_ttc/transit_data_combined.csv"))

(We couldn't take the screenshot because we lost access)

2. After loading the data in spark data frame, we tried to query the dataset to gain insights. Below are some queries and their outputs.

1. Maximum incident count by location:

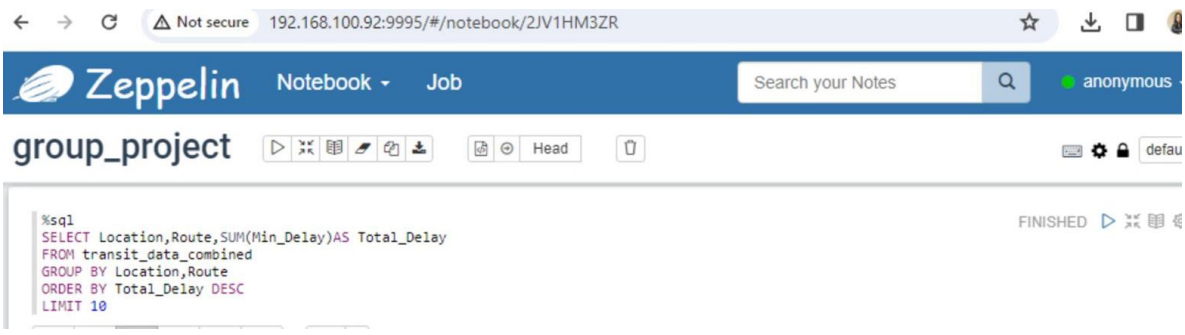


Table view for 2022:	Table view for 2023:

Location	Incident_Count
KENNEDY STATION	1384
KIPLING STATION	1276
PIONEER VILLAGE STA	1138
FINCH STATION	1101
EGLINTON STATION	1056
WILSON STATION	839
EGLINTON WEST STA	785
SCARBOROUGH CEN	698
WARDEN STATION	683
PAPE STATION	658

Location	Incident_Count
KENNEDY STATION	1231
KIPLING STATION	1065
WILSON STATION	976
FINCH STATION	924
EGLINTON STATION	902
PIONEER VILLAGE S	874
PAPE STATION	671
WARDEN STATION	611
SCARBOROUGH CE	596
EGLINTON WEST ST	523

Observations:

- Kennedy station and Kipling station had maximum incidents of delay for both the years.
- There is a drop in the overall rate of incidents.
- Pape station, Wilson station saw a rise in incidents. On the other hand, Pioneer Village Station, Eglinton West station saw a decrease in the number of incidents, which is a good thing.

2. Top 10 routes with maximum delay

```
%sql
SELECT Location,Route,SUM(Min_Delay)AS Total_Delay
FROM transit_data_combined
GROUP BY Location,Route
ORDER BY Total_Delay DESC
LIMIT 10
```

FINISHED

Observations:

- Routes with maximum delays are: Lawrence West Station, Pape Station, Eglinton Station.

3. Months with maximum delay

```
%sql
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    SUM(Min_delay) AS Total_Minutes_Delay
FROM
    transit_data_combined
WHERE
    YEAR(Date) = 2022
GROUP BY
    YEAR(Date),
    MONTH(Date)
ORDER BY
    Year,
    Month
```

FINISHED

Table view for 2022:	Table view for 2023:

Month	Total_Minutes_Delay
1	116637
8	109327
7	109181
10	108666
12	103668
9	101285
2	100488
6	97007
5	93103
11	93010
4	76467
3	71419

Month	Total_Minutes_Delay
8	111660
9	107798
10	107482
7	104142
11	98253
12	96306
3	91890
1	91007
6	87607
2	87417
5	82082
4	72534

Observation:
- We can observe delay increases at the end months of the year from July to December.
- This can be due to holiday season, weather conditions and traffic conditions.

4. Top 10 Incident Types and The Frequency

Table view for 2022:

Incident	Frequency
Operations - Operat	19583
Mechanical	16465
Collision - TTC	3511
Security	3373
Utilized Off Route	3240
General Delay	3217
Diversion	2881
Emergency Services	2416
Cleaning - Unsanitar	1561
Investigation	905

Table view for 2023:

Incident	Frequency
Mechanical	19228
Operations - Operator	11359
Security	4803
Collision - TTC	3909
Diversion	3805
General Delay	3199
Emergency Services	3016
Utilized Off Route	2361
Cleaning - Unsanitary	2152
Investigation	1247

Observation:

- Incidents with Maximum Frequency: Operator, Mechanical, Collision, Security
- Incidents with Minimum Frequency: Cleaning and Investigation.

- It can be observed that operations were improved which led to decrease in its incident frequency. However, mechanical disruptions increased from 2022 to 2023, which needs to be looked into by the management.

5. Max Incident Count by Station

```
%sql
SELECT Location, COUNT(Incident) AS Incident_Count
FROM transit_data_combined
WHERE Date >= '2022-01-01' AND Date <= '2022-12-31'
GROUP BY Location
ORDER BY Incident_Count DESC
LIMIT 10
```

FINISHED

- For both years (2022 & 2023), the station with maximum incident counts are Kennedy Station, Kipling Station
- Both of these stations are terminal stations on Line 2.

6. Maximum delay by vehicle number

```
%sql
SELECT Vehicle, SUM(Min_delay) AS Total_Delay
FROM transit_data_combined
WHERE Vehicle != 0 AND YEAR(Date) = 2022
GROUP BY Vehicle
ORDER BY Total_Delay DESC
LIMIT 10
```

Table view for 2022:

Vehicle	Total_Delay
8418	2538
8201	2481
8456	2231
8409	1854
8404	1721
1103	1699
8562	1684
8502	1672
3483	1667
9209	1649

Table view for 2023:

Vehicle	Total_Delay
8053	2675
8301	2204
3358	2031
3513	1940
3537	1904
3213	1845
8568	1792
8594	1754
8431	1701
8079	1646

- Vehicle number with maximum delay in 2022: 8418, 8456, 8201 (42.18, 41.21, 37.11 HRS)
- Vehicle number with maximum delay in 2023: 8053, 8301, 3358 (44.35, 36.44, 33.51 HRS)

References:

Iamsuzank. (2023, October 16). TTC delay analysis 2022 to 2023. Kaggle. https://www.kaggle.com/code/iamsuzank/ttc-delay-analysis-2022-to-2023/notebook