Project Instructions:

Write three SQL queries to answer the following questions:

1. What are the most popular transport types, measured by the total number of journeys? The output should contain two columns, 1) `JOURNEY_TYPE` and 2) `TOTAL_JOURNEYS_MILLIONS`, and be sorted by the second column in descending order. Save the query as `most_popular_transport_types`.

2. Which five months and years were the most popular for the Emirates Airline? Return an output containing `MONTH`, `YEAR`, and `JOURNEYS_MILLIONS`, with the latter rounded to two decimal places and aliased as `ROUNDED_JOURNEYS_MILLIONS`. Exclude null values and save the result as `emirates_airline_popularity`.

3. Find the *five years* with the lowest volume of `Underground & DLR` journeys, saving as `least_popular_years_tube`. The results should contain the columns `YEAR`, `JOURNEY_TYPE`, and `TOTAL_JOURNEYS_MILLIONS`.

Three SQL cells have been created for you in the workbook. To access the Snowflake database, you will need to select data using the syntax `FROM TFL.JOURNEYS` (ensure you use upper-case).

Note: Please also ensure that you do not change the names of the DataFrames that the three query results will be saved as - creating new cells in the workbook will rename the DataFrame (see image below). Make sure that your final solutions use the names provided: `most_popular_transport_types`, `emirates_airline_popularity`, and `least_popular_years_tube`.

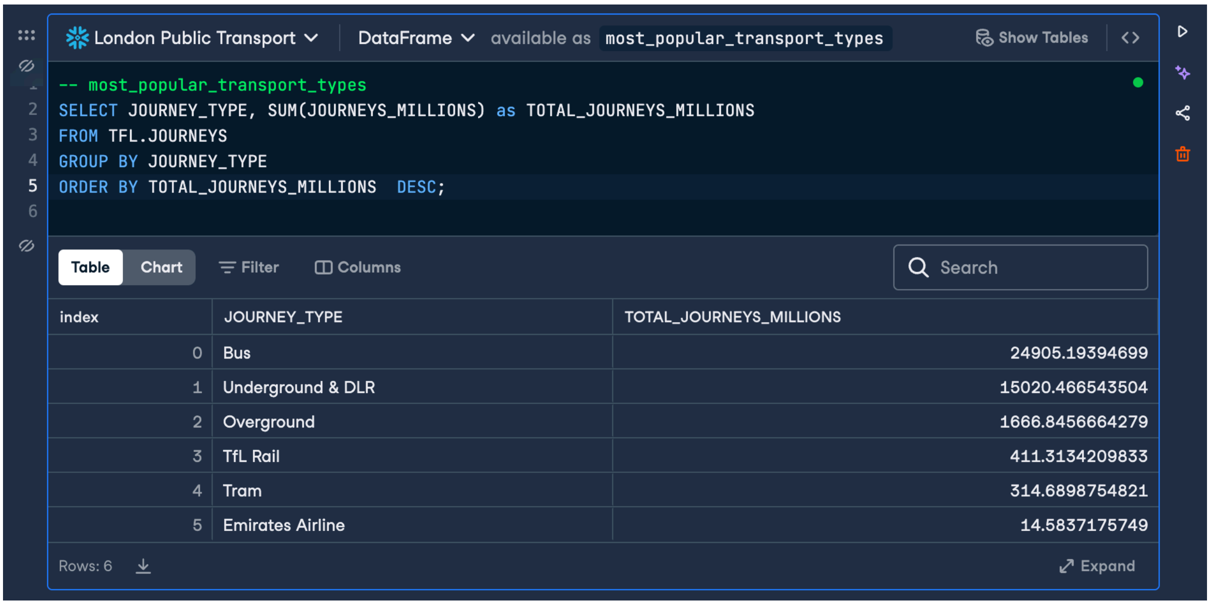
1 >>>>

SELECT JOURNEY_TYPE, SUM(JOURNEYS_MILLIONS) as TOTAL_JOURNEYS_MILLIONS

FROM TFL.JOURNEYS

GROUP BY JOURNEY_TYPE

ORDER BY TOTAL_JOURNEYS_MILLIONS DESC;



```
-- most_popular_transport_types
SELECT JOURNEY_TYPE, SUM(JOURNEYS_MILLIONS) as TOTAL_JOURNEYS_MILLIONS
FROM TFL.JOURNEYS
GROUP BY JOURNEY_TYPE
ORDER BY TOTAL_JOURNEYS_MILLIONS  DESC;
```

index	JOURNEY_TYPE	TOTAL_JOURNEYS_MILLIONS
0	Bus	24905.19394699
1	Underground & DLR	15020.466543504
2	Overground	1666.8456664279
3	TfL Rail	411.3134209833
4	Tram	314.6898754821
5	Emirates Airline	14.5837175749

Rows: 6

2 >>>>

SELECT MONTH, YEAR, ROUND(JOURNEYS_MILLIONS,2) as ROUNDED_JOURNEYS_MILLIONS

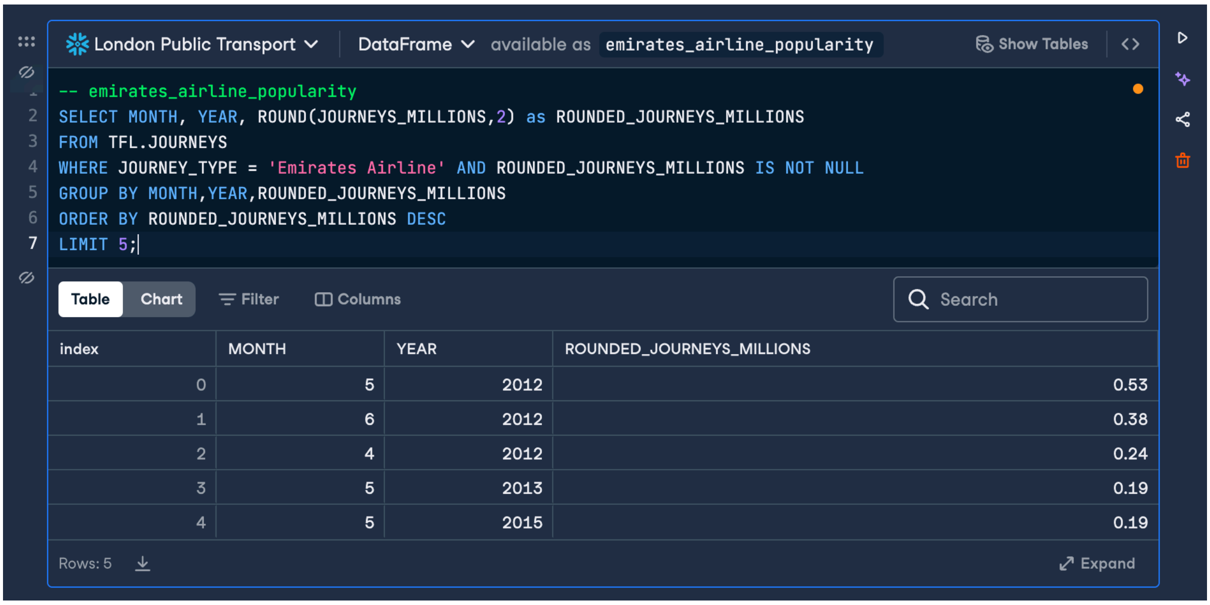
FROM TFL.JOURNEYS

WHERE JOURNEY_TYPE = 'Emirates Airline' AND ROUNDED_JOURNEYS_MILLIONS IS NOT NULL

GROUP BY MONTH,YEAR,ROUNDED_JOURNEYS_MILLIONS

ORDER BY ROUNDED_JOURNEYS_MILLIONS DESC

LIMIT 5;



```sql
-- emirates_airline_popularity
SELECT MONTH, YEAR, ROUND(JOURNEYS_MILLIONS,2) as ROUNDED_JOURNEYS_MILLIONS
FROM TFL.JOURNEYS
WHERE JOURNEY_TYPE = 'Emirates Airline' AND ROUNDED_JOURNEYS_MILLIONS IS NOT NULL
GROUP BY MONTH,YEAR,ROUNDED_JOURNEYS_MILLIONS
ORDER BY ROUNDED_JOURNEYS_MILLIONS DESC
LIMIT 5;
```

index	MONTH	YEAR	ROUNDED_JOURNEYS_MILLIONS
0	5	2012	0.53
1	6	2012	0.38
2	4	2012	0.24
3	5	2013	0.19
4	5	2015	0.19

3 >>>>

SELECT YEAR, JOURNEY_TYPE, SUM(JOURNEYS_MILLIONS) as TOTAL_JOURNEYS_MILLIONS

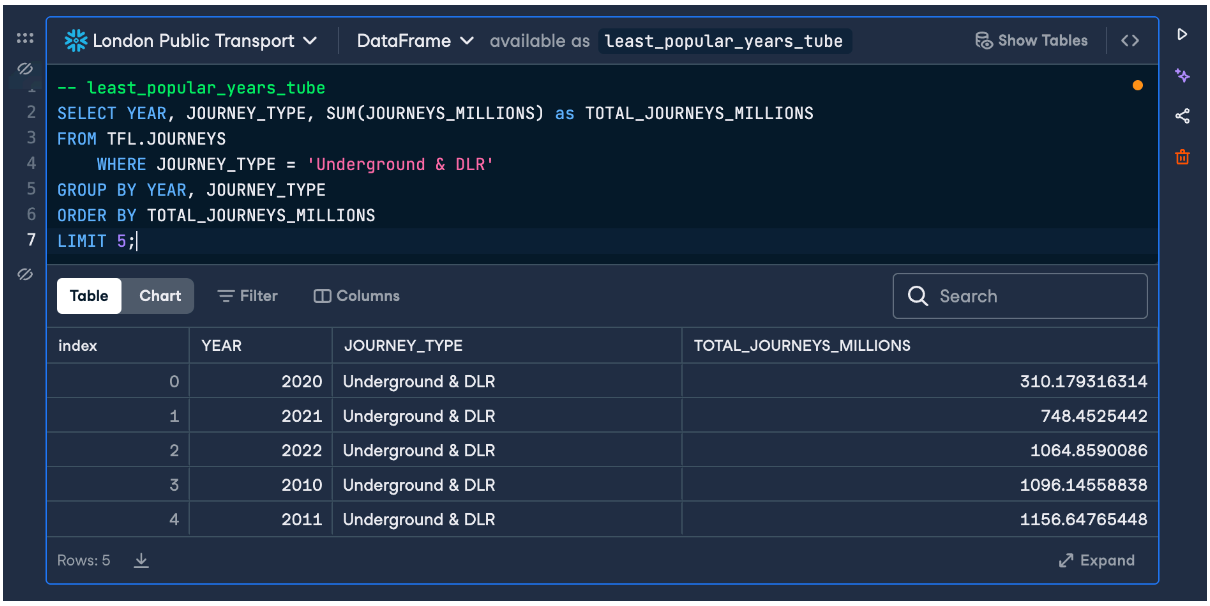
FROM TFL.JOURNEYS

WHERE JOURNEY_TYPE = 'Underground & DLR'

GROUP BY YEAR, JOURNEY_TYPE

ORDER BY TOTAL_JOURNEYS_MILLIONS

LIMIT 5;



```sql
-- least_popular_years_tube
SELECT YEAR, JOURNEY_TYPE, SUM(JOURNEYS_MILLIONS) as TOTAL_JOURNEYS_MILLIONS
FROM TFL.JOURNEYS
    WHERE JOURNEY_TYPE = 'Underground & DLR'
GROUP BY YEAR, JOURNEY_TYPE
ORDER BY TOTAL_JOURNEYS_MILLIONS
LIMIT 5;
```

index	YEAR	JOURNEY_TYPE	TOTAL_JOURNEYS_MILLIONS
0	2020	Underground & DLR	310.179316314
1	2021	Underground & DLR	748.4525442
2	2022	Underground & DLR	1064.8590086
3	2010	Underground & DLR	1096.14558838
4	2011	Underground & DLR	1156.64765448