

# MOTOR VEHICLE COLLISION- SQL QUERIES for VISUALIZATION

## 1. How many accidents occurred in NYC, Austin, and Chicago?

SQL Query:-

```
SELECT COUNT (*) AS Number_of_Accidents, d.Source
FROM Fct_Accidents f
Join Dim_Source d ON d.Source_SK = f.Source_SK
GROUP BY d.Source;
```

```
SQL Queries.sql - L.C (APEX)yugal (599)*
/*How many accidents occurred in NYC, Austin and Chicago?*/
Select Count(*) as No_of_Accidents, d.Source from Fct_Accidents f join Dim_Source d on d.Source_SK = f.Source_SK group by d.Source

/*Which areas in the 3 cities had the greatest number of accidents?*/
Select Count(f.Location_SK), d.Source from Fct_Accidents f join Dim_Source d on d.Source_SK = f.Source_SK group by d.Source order by f

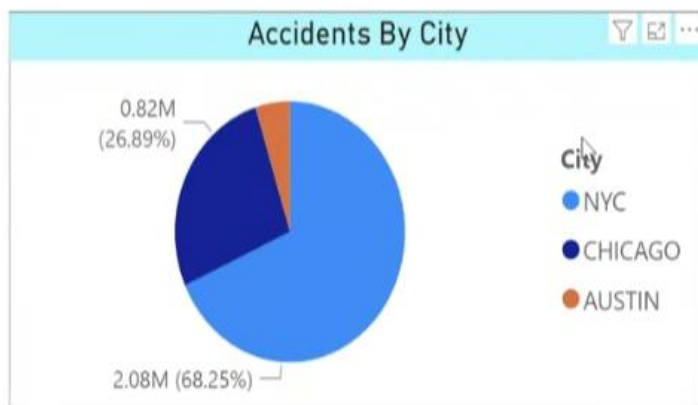
select count(STREET_NAME), SOURCE from Dim_Location where STREET_NAME like 'NA' group by SOURCE

select count(STREET_NAME), SOURCE from where STREET_NAME like 'NA' group by SOURCE

Select top 3 Count(*), l.STREET_NAME from Fct_Accidents f
join Dim_Location l on l.Location_SK = f.Location_SK
group by l.STREET_NAME, l.SOURCE
order by count(*) desc

/*How many accidents resulted in just injuries?*/
select count(*) as Accidents_Injuries from Fct_Accidents f where f.Total_Killed =0 and f.Total_Injured >0
```

No of Accidents	Source
147750	AUSTIN
817723	CHICAGO
2075427	NYC



- Which areas in the 3 cities had the greatest number of accidents? (Top 3 areas in each city).

SQL Query:-

```
WITH RankedAccidents AS (
    SELECT Dim_Location.STREET_NAME, Dim_Location.SOURCE,
           COUNT(Fct_Accidents.Accidents_FCT_SK) AS Total_Accidents,
           ROW_NUMBER() OVER (PARTITION BY Dim_Location.SOURCE ORDER BY
                               COUNT(Fct_Accidents.Accidents_FCT_SK) DESC) AS Rank
    FROM Fct_Accidents
    JOIN Dim_Location ON Fct_Accidents.Location_SK = Dim_Location.Location_SK
    GROUP BY Dim_Location.STREET_NAME, Dim_Location.SOURCE
) SELECT STREET_NAME, SOURCE, Total_Accidents FROM RankedAccidents
   WHERE Rank <= 3;
```

121 %

Results Messages

	STREET_NAME	SOURCE	Total_Accidents
1	NA	NYC	440569
2	BROADWAY	NYC	19926
3	BELT PARKWAY	NYC	18139
4	IH0035	AUSTIN	24841
5	US0183	AUSTIN	12105
6	SL0001	AUSTIN	9023
7	WESTERN AVE	CHICAGO	22319
8	PULASKI RD	CHICAGO	19695
9	CICERO AVE	CHICAGO	18335

Top N Contributing Factors			
	DRIVER I...	UNABLE ...	OTHER
	0.45M	0.43M	0.34M
	FOLLOWED TO...	FAILI...	FATI...
	0.21M		
	FAILED TO DRI...	UNSA...	BAC...
0M			

### 3. How many accidents resulted in just injuries?

#### ➤ Overall

SQL Query:-

```
SELECT COUNT (*) AS Accidents_Injuries
FROM Fct_Accidents f
WHERE f.Total_Killed =0 and f.Total_Injured >0
```

#### ➤ By City

SQL Query:-

```
SELECT d.Source, COUNT (*) AS Accidents
FROM Fct_Accidents f
JOIN Dim_Source d ON d.Source_SK = f.Source_SK
WHERE f.Total_Killed =0 and f.Total_Injured >0
GROUP BY d.Source
```

**OR**

```
SELECT Source, City_Accidents_Injuries, Overall_Accidents_Injuries
FROM (
SELECT d.Source,
      COUNT (*) AS City_Accidents_Injuries,
      SUM (COUNT (*)) OVER () AS Overall_Accidents_Injuries
FROM Fct_Accidents f
JOIN Dim_Source d ON f.Source_SK = d.Source_SK
WHERE f.Total_Killed = 0 AND f.Total_Injured > 0
GROUP BY d.Source) AS Sub;
```

The screenshot shows a SQL IDE with three queries and their results. The first query counts accidents with injuries. The second query counts accidents by city. The third query counts pedestrian accidents.

```
/*How many accidents resulted in just injuries?*/
select count(*) as Accidents_Injuries from Fct_Accidents f where f.Total_Killed =0 and f.Total_Injured >0

/*How many accidents resulted in just injuries?*/
select d.Source,count(*) as Accidents from Fct_Accidents f
join Dim_Source d on d.Source_SK = f.Source_SK
where f.Total_Killed =0 and f.Total_Injured >0
group by d.Source

/*How often are pedestrians involved in accidents?*/
```

Results:

Accidents_Injuries
649813

Source	Accidents
AUSTIN	64770
CHICAGO	111813
NYC	473430



#### 4. How often are pedestrians involved in accidents?

##### ➤ Overall

SQL Query:-

```
SELECT COUNT (*) AS Accidents_Pedestrians FROM Fct_Accidents f WHERE f.IS_PEDESTRIAN = 'Y';
```

##### ➤ By City

SQL Query:-

```
SELECT d.Source, COUNT (*) AS Accidents_Pedestrians
FROM Fct_Accidents f
JOIN Dim_Source d ON d.Source_SK = f.Source_SK
WHERE f.IS_PEDESTRIAN = 'Y'
GROUP BY d.Source
```

**OR**

```
SELECT d.Source,
COUNT (*) AS Accidents_Pedestrians,
SUM (COUNT (*)) OVER () AS Total_Accidents_Pedestrians
FROM Fct_Accidents f
JOIN Dim_Source d ON f.Source_SK = d.Source_SK
WHERE f.IS_PEDESTRIAN = 'Y'
GROUP BY d.Source;
```

```

/*How often are pedestrians involved in accidents?*/
select count(*) as Accidents_Pedestrians from Fct_Accidents f where f.IS_PEDESTRIAN = 'Y';

select d.Source, count(*) as Accidents_Pedestrians from Fct_Accidents f
join Dim_Source d on d.Source_SK = f.Source_SK
where f.IS_PEDESTRIAN = 'Y'
group by d.Source

/* When do most accidents happen?
• seasonality report*/

```

Results Messages

	Accidents_Pedestrians
1	136496

	Source	Accidents_Pedestrians
1	AUSTIN	3505
2	CHICAGO	19065
3	NYC	113926



## 5. When do most accidents happen? (Seasonality Report)

SQL Query:-

```

SELECT d.Season, COUNT (*) AS No_of_Accidents FROM Fct_Accidents f
JOIN Dim_Date d ON f.Date_SK = d.Date_SK
GROUP BY d.Season
ORDER BY COUNT (*) DESC;

```

```

SELECT Season, No_of_Accidents,
       SUM (No_of_Accidents) OVER (ORDER BY No_of_Accidents DESC) AS Cumulative_Total,
       ROUND (100.0 * No_of_Accidents / SUM (No_of_Accidents) OVER (), 2) AS Percentage_of_Total
FROM (
  SELECT d.Season, COUNT (*) AS No_of_Accidents
  FROM Fct_Accidents f
  JOIN Dim_Date d ON f.Date_SK = d.Date_SK
  GROUP BY d.Season
) AS SeasonData
ORDER BY No_of_Accidents DESC;

```

```

/* When do most accidents happen?
• seasonality report*/
select d.Season, count(*) as No_of_Accidents from Fct_Accidents f
join Dim_Date d on f.Date_SK = d.Date_SK
group by d.Season
order by count(*) desc
/*Final Project Details

```

	Season	No of Accidents
1	Fall	806512
2	Summer	784509
3	Winter	733507
4	Spring	716282



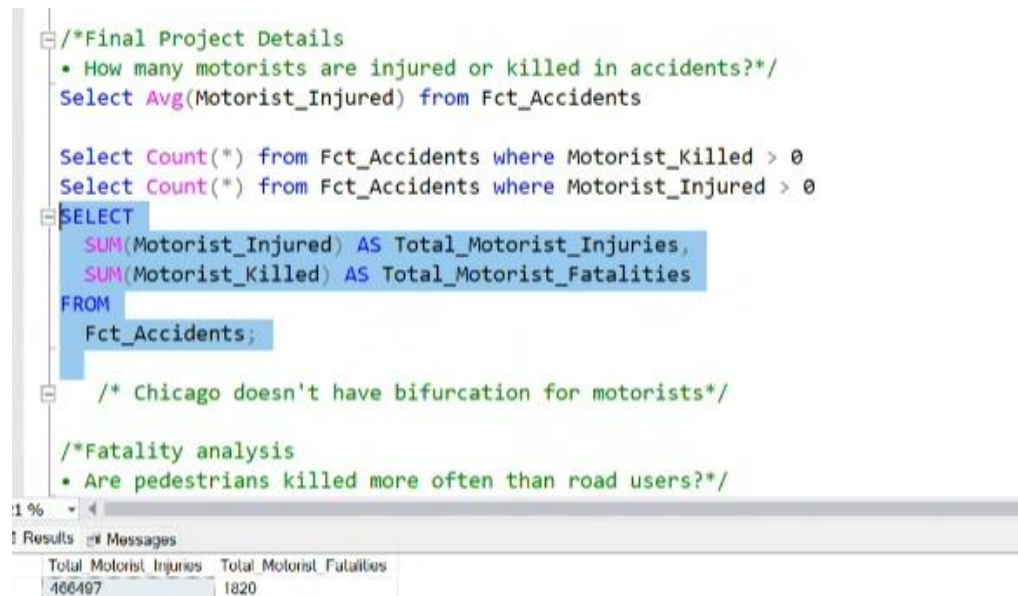
## 6. How many motorists are injured or killed in accidents?

SQL Query:-

```
SELECT
    SUM (Motorist_Injured) AS Total_Motorist_Injuries,
    SUM (Motorist_Killed) AS Total_Motorist_Fatalities
FROM Fct_Accidents;
```

```
SELECT s.Source,
    SUM (Motorist_Injured) AS Total_Motorist_Injuries,
    SUM (Motorist_Killed) AS Total_Motorist_Fatalities
FROM Fct_Accidents f
JOIN Dim_Source s ON f.Source_SK = s.Source_SK
GROUP BY S.Source; /* Chicago doesn't have bifurcation for motorists*/
```

```
SELECT s.Source,
    SUM (f.Motorist_Injured) AS Total_Motorist_Injuries,
    SUM (f.Motorist_Killed) AS Total_Motorist_Fatalities,
    SUM (SUM (f.Motorist_Injured)) OVER () AS Overall_Motorist_Injuries,
    SUM (SUM (f.Motorist_Killed)) OVER () AS Overall_Motorist_Fatalities
FROM Fct_Accidents f
JOIN Dim_Source s ON f.Source_SK = s.Source_SK
GROUP BY s.Source
ORDER BY s.Source;
```



The screenshot shows a SQL IDE with a query editor and a results pane. The query editor contains the following SQL code:

```
/*Final Project Details
• How many motorists are injured or killed in accidents?*/
Select Avg(Motorist_Injured) from Fct_Accidents

Select Count(*) from Fct_Accidents where Motorist_Killed > 0
Select Count(*) from Fct_Accidents where Motorist_Injured > 0

SELECT
    SUM(Motorist_Injured) AS Total_Motorist_Injuries,
    SUM(Motorist_Killed) AS Total_Motorist_Fatalities
FROM
    Fct_Accidents;

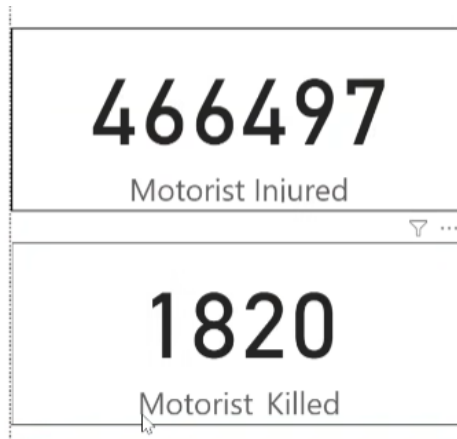
/* Chicago doesn't have bifurcation for motorists*/

/*Fatality analysis
• Are pedestrians killed more often than road users?*/
```

The results pane shows the following data:

Total_Motorist_Injuries	Total_Motorist_Fatalities
466497	1820





## 7. Which top 5 areas in 3 cities have the most fatal accidents?

SQL Query:-

```
SELECT top 5 COUNT (*), I.STREET_NAME, I.SOURCE FROM Fct_Accidents f
JOIN Dim_Location I ON I.Location_SK = f.Location_SK
GROUP BY I.STREET_NAME, I.SOURCE
HAVING I.STREET_NAME NOT LIKE 'NA'
ORDER BY COUNT (*) DESC /*Given NYC's NA COUNT of street name is ignored*/
```

SQL Query:-

```
WITH RankedAccidents AS (
    SELECT Dim_Location.STREET_NAME, Dim_Source.SOURCE,
           COUNT (Fct_Accidents.Accidents_FCT_SK) AS Total_Accidents,
           ROW_NUMBER () OVER (PARTITION BY Dim_Source.SOURCE
                                ORDER BY COUNT (Fct_Accidents.Accidents_FCT_SK) DESC) AS Rank
    FROM Fct_Accidents
    JOIN Dim_Location ON Fct_Accidents.Location_SK = Dim_Location.Location_SK
    JOIN Dim_Source ON Fct_Accidents.Source_SK = Dim_Source.Source_SK
    GROUP BY Dim_Location.STREET_NAME, Dim_Source.SOURCE
) SELECT STREET_NAME, SOURCE, Total_Accidents
FROM RankedAccidents WHERE Rank <= 5;
```



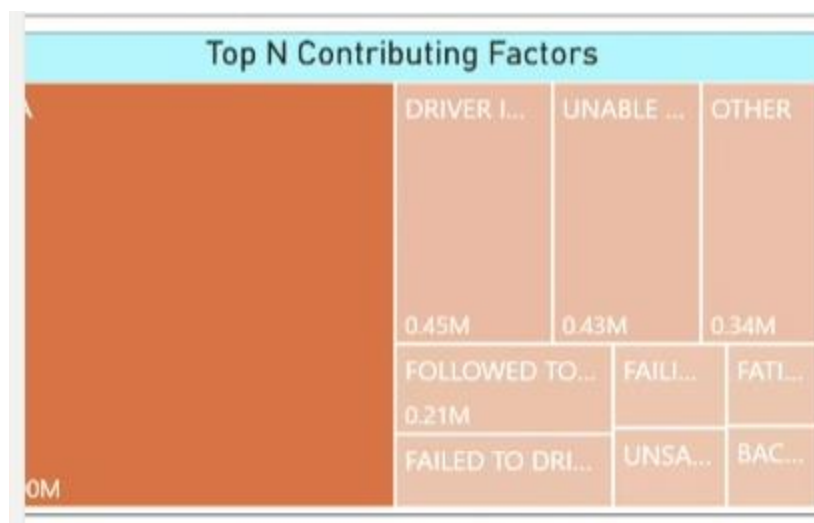
```

/*Which top 5 areas in 3 cities have the most fatal number of accidents?*/

WITH RankedAccidents AS (
    SELECT Dim_Location.STREET_NAME, Dim_Source.SOURCE,
    COUNT(Fct_Accidents.Accidents_FCT_SK) AS Total_Accidents,
    ROW_NUMBER() OVER (PARTITION BY Dim_Source.SOURCE ORDER BY COUNT(Fct_Accidents.Accidents_FCT_SK) DESC) AS Rank
    FROM Fct_Accidents
    JOIN Dim_Location ON Fct_Accidents.Location_SK = Dim_Location.Location_SK
    JOIN Dim_Source ON Fct_Accidents.Source_SK = Dim_Source.Source_SK
    GROUP BY Dim_Location.STREET_NAME, Dim_Source.SOURCE
) SELECT STREET_NAME, SOURCE, Total_Accidents
FROM RankedAccidents
WHERE Rank <= 5;

```

	STREET_NAME	SOURCE	Total_Accidents
1	IH0035	AUSTIN	24841
2	US0183	AUSTIN	12105
3	SI 0001	AUSTIN	9023
4	US0290	AUSTIN	5206
5	SH0071	AUSTIN	3790
6	NA	NYC	440569
7	BROADWAY	NYC	19826
8	BELT PARKWAY	NYC	18139
9	ATLANTIC AVENUE	NYC	17801
10	3 AVENUE	NYC	14107
11	WESTERN AVE	CHICAGO	22319
12	PULASKI RD	CHICAGO	19695
13	CICERO AVE	CHICAGO	18335
14	ASHLAND AVE	CHICAGO	17755
15	HALSTED ST	CHICAGO	15851



## 8. Time-based analysis of accidents (Time of the day, day of the week, weekdays or weekends).

### ➤ Day of the Week

```

SELECT d.Day, COUNT (*) AS No_of_Accidents
FROM Fct_Accidents f
JOIN Dim_Date d ON f.Date_SK = d.Date_SK
GROUP BY d.Day
ORDER BY No_of_Accidents;

```

➤ Weekend vs. Weekday Accidents

-- Weekend Accidents

```
SELECT COUNT (*) AS Weekend_Accidents  
FROM Fct_Accidents f  
JOIN Dim_Date d ON f.Date_SK = d.Date_SK  
WHERE d.Day > 5;
```

-- Weekday Accidents

```
SELECT COUNT (*) AS Weekday_Accidents  
FROM Fct_Accidents f  
JOIN Dim_Date d ON f.Date_SK = d.Date_SK  
WHERE d.Day <= 5;
```

➤ Hour of the Day

```
SELECT d.Hour, COUNT (*) AS No_of_Accidents  
FROM Fct_Accidents f  
JOIN Dim_Time d ON f.Time_SK = d.Time_SK  
GROUP BY d.Hour  
ORDER BY d.Hour;
```

SQL\_Queries.sql - I...C (APEX\yugal (59))\* - X

```

select d.[Day], count(*) as No_of_Accidents from Fct_Accidents f
join Dim_Date d on f.Date_SK = d.Date_SK
group by d.[Day]
order by No_of_Accidents

select count(*) as Weekend_Accidents from Fct_Accidents f
join Dim_Date d on f.Date_SK = d.Date_SK
where d.[Day] > 5

select count(*) as Weekday_Accidents from Fct_Accidents f
join Dim_Date d on f.Date_SK = d.Date_SK
where d.[Day] <= 5

select d.[Hour], count(*) as No_of_Accidents from Fct_Accidents f
join Dim_Time d on f.Time_SK = d.Time_SK
group by d.[Hour]

```

121 %

Results Messages

	Day	No. of Accidents
1	1	368738
2	7	421939
3	2	429002
4	4	440445
5	3	443249
6	5	449692
7	6	487835

	Weekend_Accidents
1	909774

	Weekday_Accidents
1	2131126

	Hour	No. of Accidents
1	0	88228
2	1	54356
3	2	44918
4	3	37557
5	4	38795
6	5	43249
7	6	68284
8	7	104279

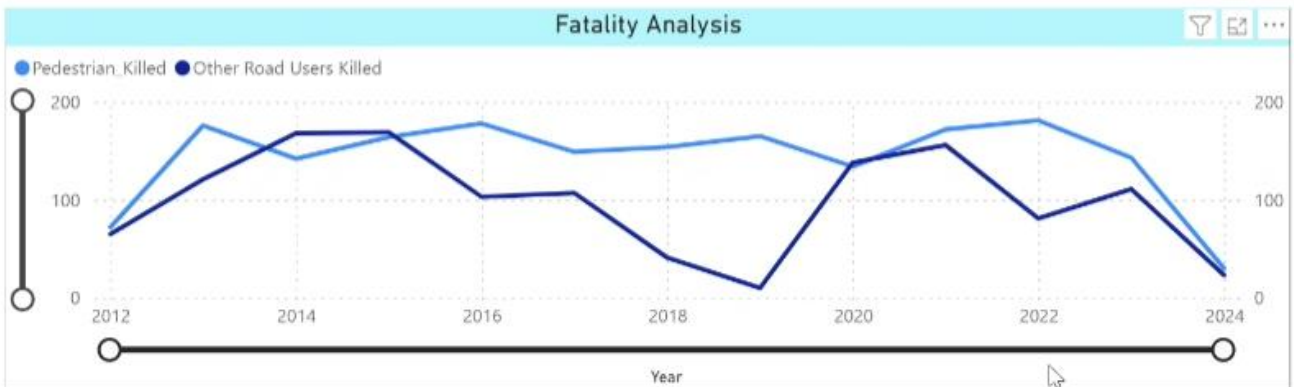
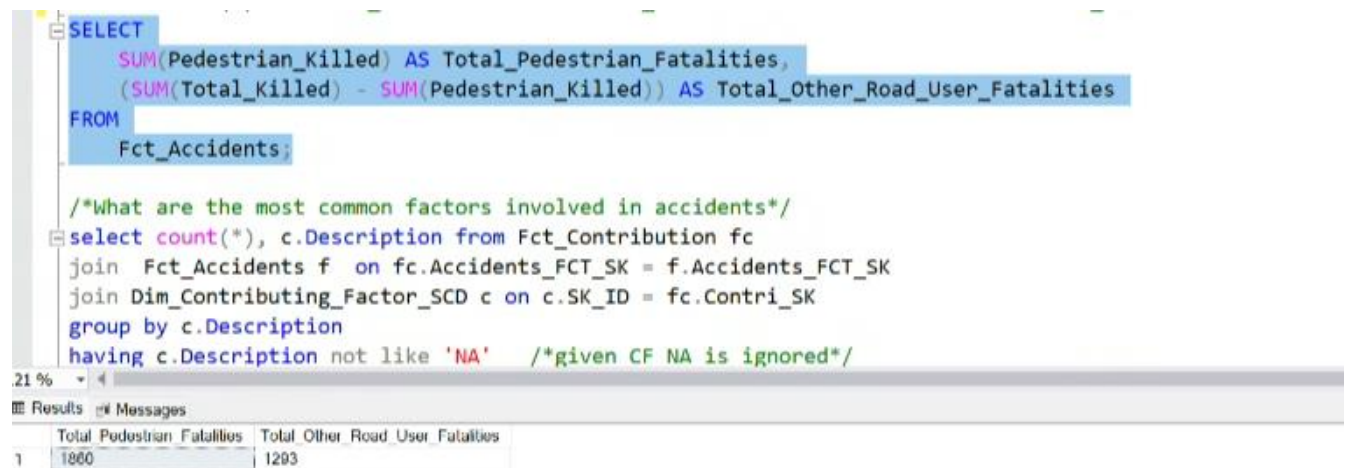
Query executed successfully.

USE FILTERS ON VISUALIZATION

## 9. Are pedestrians killed more often than road users? (Fatality analysis)

SELECT

```
SUM (Pedestrian_Killed) AS Total_Pedestrian_Fatalities,
(SUM (Total_Killed) - SUM (Pedestrian_Killed)) AS Total_Other_Road_User_Fatalities
FROM Fct_Accidents;
```



## 10. What are the most common factors involved in accidents?

SELECT

```
COUNT (*) Number, c.Description AS Most_Common_Factors FROM Fct_Contribution fc
JOIN Fct_Accidents f ON fc.Accidents_FCT_SK = f.Accidents_FCT_SK
JOIN Dim_Contributing_Factor_SCD c ON c.SK_ID = fc.Contri_SK
GROUP BY c.Description
ORDER BY COUNT (*) DESC;
```

```

select TOP 10 count(*), c.Description as Most_Common_Factors from Fct_Contribution fc
join Fct_Accidents f on fc.Accidents_FCT_SK = f.Accidents_FCT_SK
join Dim_Contributing_Factor_SCD c on c.SK_ID = fc.Contri_SK
group by c.Description
/*Top 10*/
order by count(*) desc

```

	(No column name)	Most_Common_Factors
1	1004529	NA
2	451447	DRIVER INATTENTION
3	425838	UNABLE TO DETERMINE
4	338755	OTHER
5	214886	FOLLOWED TOO CLOSELY
6	175177	FAILED TO DRIVE IN SINGLE LANE
7	104173	FAILING TO YIELD RIGHT-OF-WAY
8	98381	UNSAFE SPEED
9	83129	FATIGUED OR ASLEEP
10	82558	BACKED WITHOUT SAFETY



/\* ADDITIONAL REQUIREMENT\*/

(Using Austin and NYC datasets, Create a visualization to show the number of Accident that involved more than 2 vehicles. Show this data as a comparison between these 2 cities).

SELECT

```

Count (Vehicle_Fct_Sk) AS Number_of_Vehicle, s.Source
FROM Fct_Accidents a
JOIN Fct_Vehicle f ON f.Accidents_FCT_SK = a.Accidents_FCT_SK
JOIN Dim_Source s ON a.Source_SK = s.Source_SK
WHERE f.Vehicle_Count >2
GROUP BY s.Source

```

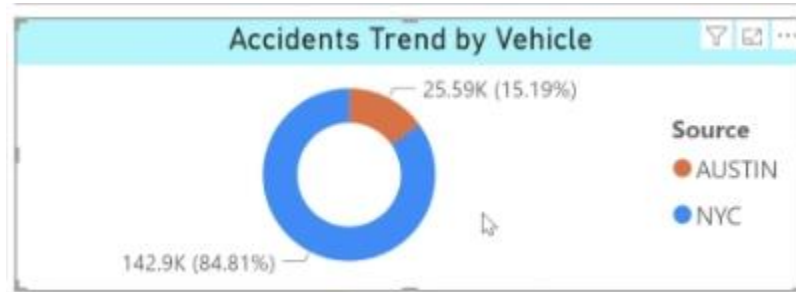
```
/*Accidents that involved more than 2 vehicles *Austin and NYC* */
```

```
Select Count(a.Accidents_FCT_SK) as No_Of_Accidents, s.Source from Fct_Accidents a join  
Fct_Vehicle f on f.Accidents_FCT_SK = a.Accidents_FCT_SK  
join Dim_Source s on a.Source_SK = s.Source_SK  
where f.Vehicle_Count >2  
Group by s.Source
```

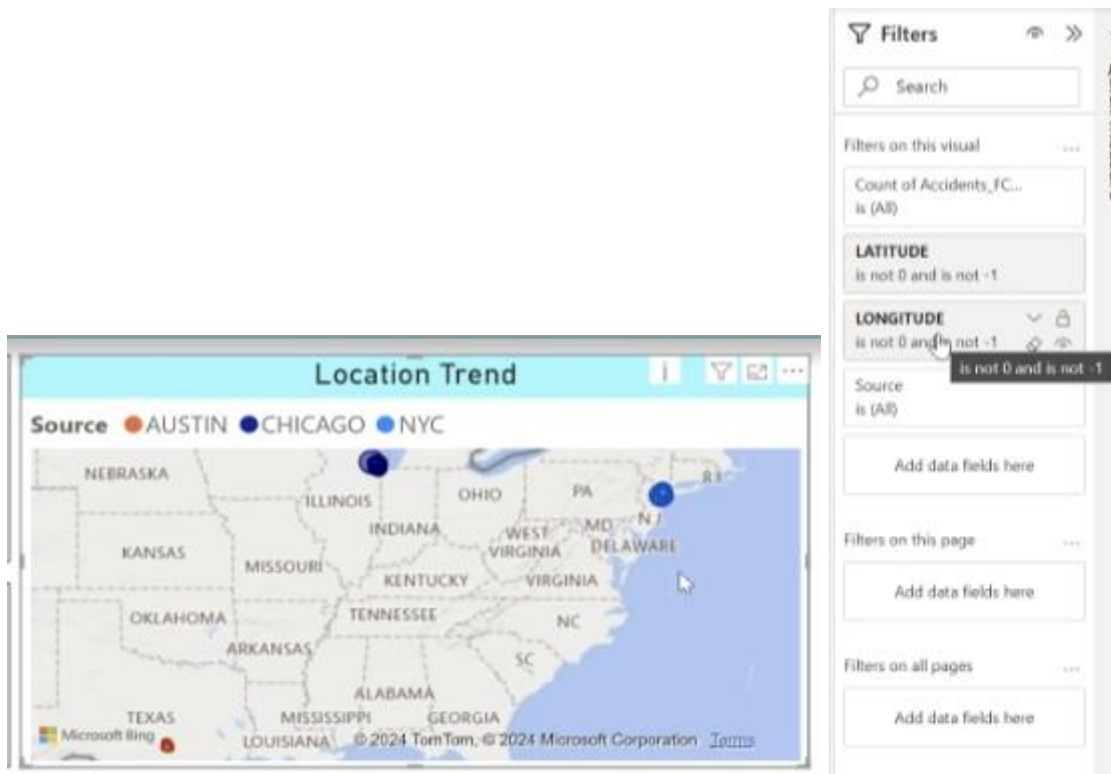
1% 4

Results Messages

No. Of Accidents	Source
48845	AUSTIN
270784	NYC



## /\* LOCATION TREND\*/



- Ignoring 0 and 1 values for latitude longitude as they are outlier.
- Latitude longitude combination (0,0) is present in New York.
- Latitude longitude combination (-1,-1) is to replace missing values.

```
/*Location Trend Observations*/

Select Count(*), S.Source from Fct_Accidents a join
Dim_Location l on a.Location_SK = l.Location_SK
join
Dim_Source s on a.Source_SK = s.Source_SK
where LATITUDE = 0 and LONGITUDE = 0
group by S.Source
```

121 % - 4

Results Messages

	(No column name)	Source
1	49	CHICAGO
2	4360	NYC



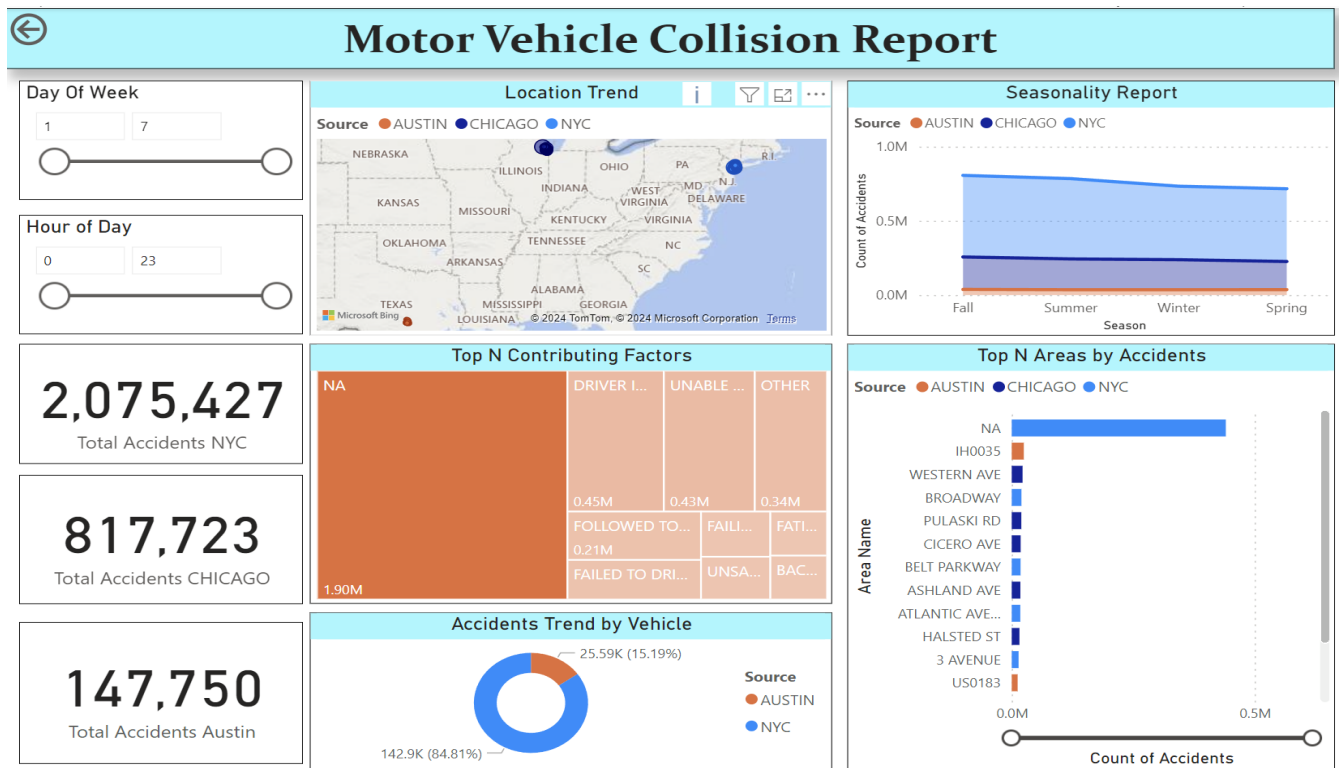
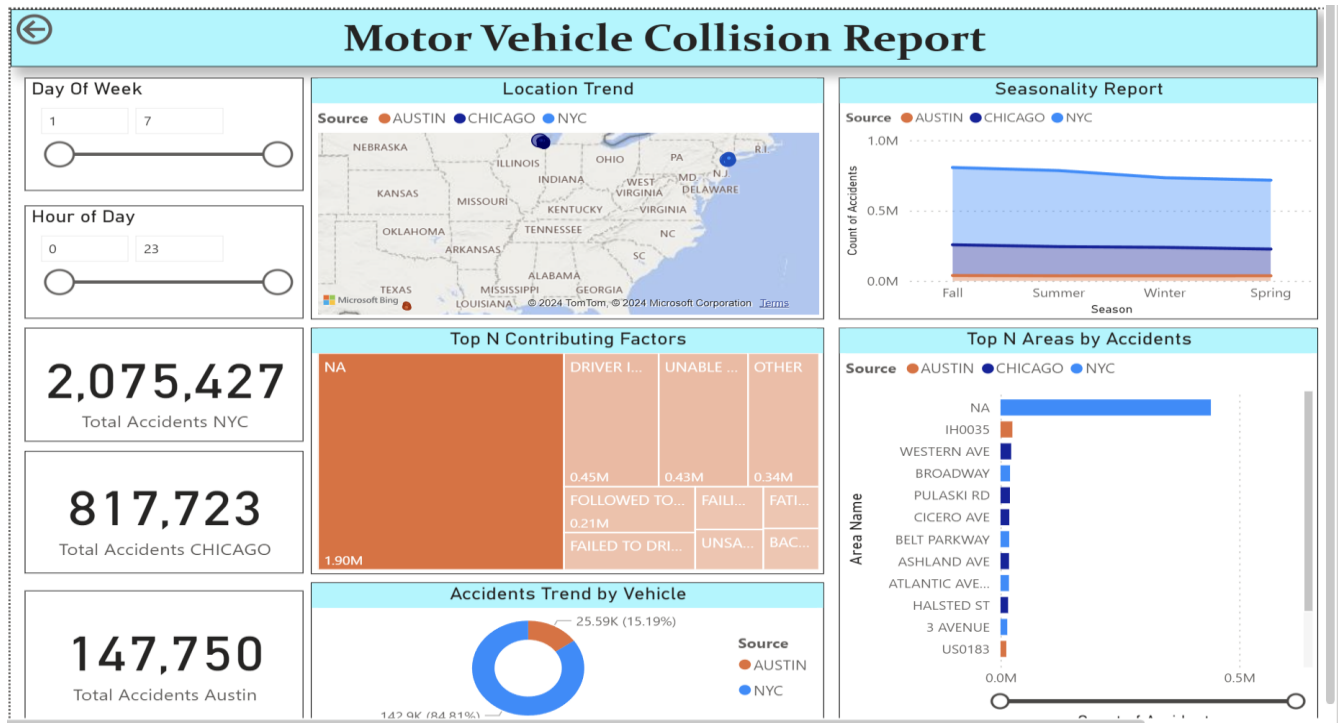
```
Select Count(*) as Count_Of_Accidents, S.Source from Fct_Accidents a join  
Dim_Location l on a.Location_SK = l.Location_SK  
join  
Dim_Source s on a.Source_SK = s.Source_SK  
where LATITUDE = -1 and LONGITUDE = -1  
group by s.Source
```

121 %

Results Messages

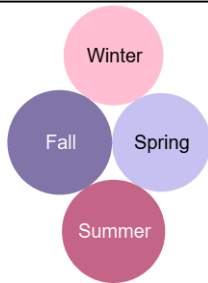
	Count_Of_Accidents	Source
1	2243	AUSTIN
2	5615	CHICAGO
3	233626	NYC

# PowerBI VISUALIZATION

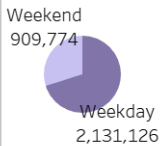


# Tableau VISUALIZATION

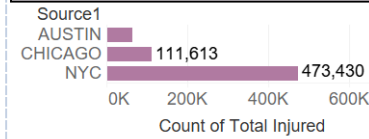
## Seasonality Report Most Accidents



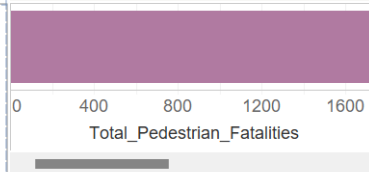
## Weekend vs. Weekday Visualizati



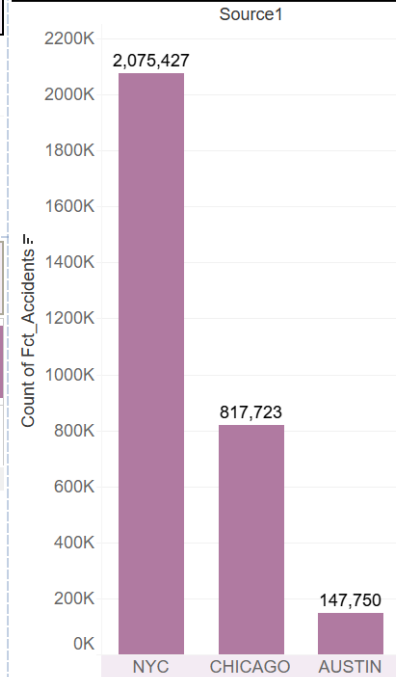
## Accidents Resulted Just Injured



## Pedestrians are killed more often than road users



## Number of Accidents



Weekend Indicat..  
Weekday  
Weekend

Count of Acciden..  
3,040,900

Season  
Fall  
Spring  
Summer  
Winter

## Top 3 Areas Each City

Street Name	AUSTIN	Source	CHICAGO	NYC
WESTERN ..		22,319		
PULASKI RD		19,695		
CICERO AVE		18,335		
ASHLAND A..		17,755		
HALSTED ST		15,851		
KEDZIE AVE		14,202		
BROADWAY		4,206	19,926	
IH0035	24,841			
BELT PARK..			18,139	
ATLANTIC A..			17,601	

## Pedestrians Involved in Accidents

Source1	Count of Accidents FCT SK
AUSTIN	3,505
CHICAGO	19,065
NYC	113,926

Count of Accidents FCT SK  
4,206 24,841

Count of Fct\_Accidents  
2 384,440

## Top\_5\_Areas\_Each\_City

Street Name	NYC	CHICAGO
BROADWAY		
IH0035		
NA		
PULASKI RD		
WESTERN AVE		

## Hour Based Accident Analysis

