

Project SQL: Chicago traffic crashes

– Original query create table

```
DROP TABLE IF EXISTS crashes;
CREATE TABLE crashes (
    crash_id int PRIMARY KEY,
    crash_date timestamp,
    posted_speed_limit varchar(2),
    traffic_control_device varchar(50),
    device_condition varchar(50),
    weather_condition varchar(25),
    lighting_condition varchar(50),
    first_crash_type varchar(100),
    traffic_way_type varchar(100),
    lane_count varchar(5),
    alignment varchar(25),
    roadway_surface_condition varchar(25),
    road_defect varchar(50),
    report_type varchar(50),
    crash_type varchar(100),
    hit_and_run varchar(1),
    damage varchar(25),
    date_police_notified timestamp,
    primary_cause varchar(100),
    secondary_cause varchar(100),
    street_direction varchar(1),
    street_name varchar(50),
    statement_taken varchar(1),
    work_zone varchar(1),
    work_zone_type varchar(25),
    workers_present varchar(1),
    number_unit varchar(2),
    most_severe_injury varchar(50),
    injuries_total varchar(2),
    injuries_fatal varchar(2),
    injuries_incapacitated varchar(2),
    injuries_non_incapacitated varchar(2),
    injuries_reported_not_evident varchar(2),
    crash_hour varchar(2),
```

```
    crash_day_of_week varchar(9),
    crash_month varchar(2),
    latitude varchar(25),
    longitude varchar(25),
    crash_location varchar(75)
);
```

— Create columns in pgadmin4 then import cvs file

-- Create Crashes Table

DROP TABLE IF EXISTS crashes;

CREATE TABLE crashes (

```
    crash_id int PRIMARY KEY,
    crash_date timestamp,
    posted_speed_limit varchar(2),
    traffic_control_device varchar(50),
    device_condition varchar(50),
    weather_condition varchar(25),
    lighting_condition varchar(50),
    first_crash_type varchar(100),
    traffic_way_type varchar(100),
    lane_count varchar(5),
    alignment varchar(25),
    roadway_surface_condition varchar(25),
    road_defect varchar(50),
    report_type varchar(50),
    crash_type varchar(100),
    hit_and_run varchar(1),
    damage varchar(25),
    date_police_notified timestamp,
    primary_cause varchar(100),
    secondary_cause varchar(100),
    street_direction varchar(1),
    street_name varchar(50),
    number_unit varchar(2),
    most_severe_injury varchar(50),
    injuries_total varchar(2),
    injuries_fatal varchar(2),
    injuries_incapacitated varchar(2),
    injuries_non_incapacitated varchar(2),
    injuries_reported_not_evident varchar(2),
```

```

        crash_hour varchar(2),
        crash_day_of_week varchar(9),
        crash_month varchar(2),
        latitude varchar(25),
        longitude varchar(25),
        crash_location varchar(75),
    );

```

```

COPY crashes
FROM
'/Users/rusamijan/Desktop/Chicago_traffic_crashes-main/update
csv/CTC_cleaned.csv'
DELIMITER ',' CSV HEADER;

```

```

-- checking null value
SELECT *
FROM crashes
WHERE crash_date is NULL;

```

```

-- checking null value
SELECT *
FROM crashes
WHERE crash_date is NULL;

```

Data Output	Messages	Notifications
<div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> </div>		
crash_id [PK] integer	crash_date timestamp without time zone	posted_speed_limit character varying (2)
		traffic_control_device character varying (50)
		device_condition character varying

```

-- checking duplicate value
SELECT crash_id,
        crash_date,
        posted_speed_limit,
        traffic_control_device,
        device_condition,
        weather_condition,
        lighting_condition,
        first_crash_type,
        traffic_way_type
        lane_count,
        alignment,
        roadway_surface_condition,

```

```
road_defect,  
report_type,  
crash_type,  
hit_and_run,  
damage,  
date_police_notified,  
primary_cause  
secondary_cause,  
street_direction,  
street_name,  
statement_taken  
work_zone,  
work_zone_type,  
workers_present  
number_unit,  
most_severe_injury,  
injuries_total,  
injuries_fatal,  
injuries_incapacitated,  
injuries_non_incapacitated,  
injuries_reported_not_evident,  
crash_hour,  
crash_day_of_week,  
crash_month,  
latitude,  
longitude,  
crash_location,  
COUNT(*)  
FROM crashes  
GROUP BY crash_id,  
crash_date,  
posted_speed_limit,  
traffic_control_device,  
device_condition,  
weather_condition,  
lighting_condition,  
first_crash_type,  
traffic_way_type,  
lane_count,  
alignment,
```

roadway_surface_condition,
road_defect,
report_type,
crash_type,
hit_and_run,
damage,
date_police_notified,
primary_cause,
secondary_cause,
street_direction,
street_name,
statement_taken,
work_zone,
work_zone_type,
workers_present,
number_unit,
most_severe_injury,
injuries_total,
injuries_fatal,
injuries_incapacitated,
injuries_non_incapacitated,
injuries_reported_not_evident,
crash_hour,
crash_day_of_week,
crash_month,
latitude,
longitude,
crash_location
HAVING COUNT(*)>1;

Query Query History

```
1  -- checking duplicate value
2  SELECT crash_id,
3         crash_date,
4         posted_speed_limit,
5         traffic_control_device,
6         device_condition,
7         weather_condition,
8         lighting_condition,
9         first_crash_type,
10        traffic_way_type,
11        lane_count,
12        alignment,
13        roadway_surface_condition,
14        road_defect,
15        report_type,
16        crash_type,
17        hit_and_run,
18        damage,
19        date_police_notified,
20        primary_cause,
21        secondary_cause,
22        street_direction,
23        street_name,
24        statement_taken
```

Data Output					Messages	Notifications
+	+	+	+	+		
crash_id	crash_date	posted_speed_limit	traffic_control_device	device_condition		
[PK] integer	timestamp without time zone	character varying (2)	character varying (50)	character varying		

No duplicate data

— Checking database info

```
SELECT *
FROM crashes;
```

Query Query History

```
1  SELECT *
2  FROM crashes;
```

Data Output									Messages	Notifications
+	+	+	+	+	+	+	+	+		
	crash_id	crash_date	posted_speed_limit	traffic_control_device	device_condition	weather_condition	lighting_conditior			
	[PK] integer	timestamp without time zone	character varying (2)	character varying (50)	character varying (50)	character varying (25)	character varying			
1	1	2019-03-25 14:43:00	30	traffic signal	functioning properly	clear	daylight			
2	2	2018-09-05 08:40:00	30	no controls	no controls	clear	daylight			
3	3	2022-07-15 00:45:00	30	unknown	unknown	clear	darkness, lighted			
4	4	2022-07-15 18:50:00	30	traffic signal	functioning properly	clear	daylight			
5	5	2018-06-03 17:00:00	30	no controls	no controls	clear	unknown			
6	6	2022-07-24 19:23:00	25	no controls	no controls	clear	daylight			
7	7	2022-07-15 17:10:00	40	no controls	no controls	cloudy/overcast	daylight			
8	8	2017-05-05 13:00:00	30	traffic signal	functioning properly	clear	daylight			
9	9	2018-11-26 10:00:00	30	traffic signal	functioning properly	snow	daylight			
10	10	2020-07-15 11:45:00	25	traffic signal	functioning properly	clear	daylight			
11	11	2017-09-13 18:00:00	30	no controls	no controls	clear	daylight			
12	12	2022-09-16 22:00:00	35	no controls	no controls	clear	darkness, lighted			
13	13	2022-07-15 08:44:00	30	no controls	no controls	rain	daylight			
14	14	2022-07-15 16:10:00	30	stop sign/flasher	functioning properly	rain	daylight			
15	15	2022-07-15 11:55:00	35	traffic signal	not functioning	rain	daylight			
16	16	2022-07-15 13:30:00	30	no controls	no controls	rain	daylight			
17	17	2022-07-15 21:00:00	30	unknown	unknown	rain	darkness			
18	18	2018-02-24 14:37:00	25	no controls	no controls	clear	daylight			

--Descriptive statistics (MIN and MAX) for numerical columns for timeline_crash

```
SELECT MIN(crash_date) AS min_crash_date,
       MAX(crash_date) AS max_crash_date,
       MIN(posted_speed_limit) AS min_speed_limit,
       MAX(posted_speed_limit) AS max_speed_limit,
       MIN(lane_count) AS min_lane_count,
```

MAX(lane_count) AS max_lane_count,
MIN(damage) AS min_damage,
MAX(damage) AS max_damage,
MIN(date_police_notified) AS min_date_police_notified,
MAX(date_police_notified) AS max_date_police_notified,
MIN(number_unit) AS min_number_unit,
MAX(number_unit) AS max_number_unit,
MIN(injuries_total) AS min_injuries_total,
MAX(injuries_total) AS max_injuries_total,
MIN(injuries_fatal) AS min_injuries_fatal,
MAX(injuries_fatal) AS max_injuries_fatal,
MIN(injuries_incapacitated) AS min_injuries_incapacitated,
MAX(injuries_incapacitated) AS max_injuries_incapacitated,
MIN(injuries_reported_not_evident) AS
min_injuries_reported_not_evident,
MAX(injuries_reported_not_evident) AS
max_injuries_reported_not_evident,
MIN(crash_hour) AS min_crash_hour,
MAX(crash_hour) AS max_crash_hour,
MIN(crash_month) AS min_crash_month,
MAX(crash_month) AS max_crash_month,
MIN(latitude) AS min_latitude,
MAX(latitude) AS max_latitude,
MIN(longitude) AS min_longitude,
MAX(longitude) AS max_longitude,
MIN(crash_location) AS min_crash_location,
MAX(crash_location) AS max_crash_lo

Query

Query History

1

--Descriptive statistics (MIN and MAX) for numerical columns for timeline_crash

2

SELECT MIN(crash_date) AS min_crash_date,

3

MAX(crash_date) AS max_crash_date,

4

MIN(posted_speed_limit) AS min_speed_limit,

5

MAX(posted_speed_limit) AS max_speed_limit,

6

MIN(lane_count) AS min_lane_count,

7

MAX(lane_count) AS max_lane_count,

8

MIN(damage) AS min_damage,

9

MAX(damage) AS max_damage,

10

MIN(date_police_notified) AS min_date_police_notified,

11

MAX(date_police_notified) AS max_date_police_notified,

12

MIN(number_unit) AS min_number_unit,

13

MAX(number_unit) AS max_number_unit,

14

MIN(injuries_total) AS min_injuries_total,

15

MAX(injuries_total) AS max_injuries_total,

16

MIN(injuries_fatal) AS min_injuries_fatal,

17

MAX(injuries_fatal) AS max_injuries_fatal,

18

MIN(injuries_incapacitated) AS min_injuries_incapacitated,

19

MAX(injuries_incapacitated) AS max_injuries_incapacitated,

20

MIN(injuries_reported_not_evident) AS min_injuries_reported_not_evident,

21

MAX(injuries_reported_not_evident) AS max_injuries_reported_not_evident,

22

MIN(crash_hour) AS min_crash_hour,

23

MAX(crash_hour) AS max_crash_hour,

24

MIN(crash_month) AS min_crash_month,

25

MAX(crash_month) AS max_crash_month,

26

MIN(latitude) AS min_latitude,

27

MAX(latitude) AS max_latitude,

28

MIN(longitude) AS min_longitude,

29

MAX(longitude) AS max_longitude,

30

MIN(crash_location) AS min_crash_location,

31

MAX(crash_location) AS max_crash_location

32

Data Output

Messages

Notifications

	min_crash_date	max_crash_date	min_speed_limit	max_speed_limit	min_lane_count	max_lane_count	min_damage	max_damage	min_date_police_notified	max_date_police_notified
	timestamp without time zone	timestamp without time zone	text	text	text	text	text	text	timestamp without time zone	timestamp without time zone
1	2017-01-01 00:01:00	2022-12-31 23:59:00	0	9	0	99	\$500 or less	over \$1,500	2017-01-01 01:10:00	2023-01-12 17:00:00

cation
FROM crash_timeline;

-- What is the total count of recorded crashes in the complete dataset?
SELECT COUNT(*)
FROM crashes;

Query

Query History

1

2

3

4

5

6


SELECT COUNT(*)


FROM crashes;


Data Output


Messages


N















count

bigint



1

686276

Query Query History

```

1 -- create crash_timeline table (2016-2022)
2 DROP TABLE IF EXISTS crash_timeline;
3 CREATE TEMP TABLE crash_timeline AS
4 (
5     SELECT
6     *
7     FROM
8     crashes
9     WHERE
10     EXTRACT(YEAR FROM crash_date) between '2017.0' AND '2022.0'
11 );
12
13 SELECT
14 *
15 FROM crash_timeline;

```

Data Output Messages Notifications

	crash_id integer	crash_date timestamp without time zone	posted_speed_limit character varying (2)	traffic_control_device character varying (50)	device_condition character varying (50)	weather_condition character varying (25)	lighting_condition character varying (50)	first_crash_type character varying (100)	traffic_way_type character varying (100)
1	1	2019-03-25 14:43:00	30	traffic signal	functioning properly	clear	daylight	turning	one-way
2	2	2018-09-05 08:40:00	30	no controls	no controls	clear	daylight	angle	not divided
3	3	2022-07-15 00:45:00	30	unknown	unknown	clear	darkness, lighted road	angle	not divided
4	4	2022-07-15 18:50:00	30	traffic signal	functioning properly	clear	daylight	rear end	not divided
5	5	2018-06-03 17:00:00	30	no controls	no controls	clear	unknown	parked motor vehicle	one-way
6	6	2022-07-24 19:23:00	25	no controls	no controls	clear	daylight	parked motor vehicle	one-way
7	7	2022-07-15 17:10:00	40	no controls	no controls	cloudy/overcast	daylight	angle	divided - w/med
8	8	2017-05-05 13:00:00	30	traffic signal	functioning properly	clear	daylight	sideswipe opposite direction	not divided
9	9	2018-11-26 10:00:00	30	traffic signal	functioning properly	snow	daylight	angle	not divided
10	10	2020-07-15 11:45:00	25	traffic signal	functioning properly	clear	daylight	pedalcyclist	not divided
11	11	2017-09-13 18:00:00	30	no controls	no controls	clear	daylight	parked motor vehicle	parking lot
12	12	2022-09-16 22:00:00	35	no controls	no controls	clear	darkness, lighted road	rear to front	alley
13	13	2022-07-15 08:44:00	30	no controls	no controls	rain	daylight	rear end	divided - w/med
14	14	2022-07-15 16:10:00	30	stop sign/flasher	functioning properly	rain	daylight	pedestrian	four way
15	15	2022-07-15 11:55:00	35	traffic signal	not functioning	rain	daylight	turning	not divided

Due to fail to import Chicago traffic crash part 2 , the total data is not correct. But I will use what I have for analysis.

-- What is the earliest and latest date of recorded crashes?

**SELECT MAX(crash_date)AS latest_date, MIN(crash_date)AS earliest_date
FROM crashes;**

Query Query History

```

1 -- 2 What is the earliest and latest date of recorded crashes?
2 SELECT MAX(crash_date)AS latest_date, MIN(crash_date)AS earliest_date
3 FROM crashes;

```

Data Output Messages Notifications

	latest_date timestamp without time zone	earliest_date timestamp without time zone
1	2023-01-12 23:36:00	2013-03-03 16:48:00

-- What is the number of reported crashes per year?

--SELECT EXTRACT('YEAR' FROM DATE));
SELECT

```

        EXTRACT('YEAR' FROM crash_date) AS crash_year,
        count(*) AS reported_crashes
FROM
    crashes
GROUP BY
    crash_year
ORDER BY
    crash_year;

```

Query Query History

```

1  -- 3 What is the number of reported crashes per year?
2  --SELECT EXTRACT('YEAR' FROM DATE));
3  SELECT
4      EXTRACT('YEAR' FROM crash_date) AS crash_year,
5      count(*) AS reported_crashes
6  FROM
7      crashes
8  GROUP BY
9      crash_year
10 ORDER BY
11     crash_year;

```

Data Output Messages Notifications



	crash_year numeric	reported_crashes bigint
1	2013	2
2	2014	6
3	2015	9828
4	2016	44297
5	2017	83786
6	2018	118950
7	2019	117762
8	2020	92088
9	2021	108756
10	2022	108292
11	2023	2509

Found that the data in 2013 to 2016 are missing or **2017 appears to be the first year with the most complete data** so I decided to create a new table that includes only 2017 - 2022 for my analysis

```
-- Create crash_timeline table (2016-2022)
DROP TABLE IF EXISTS crash_timeline;
CREATE TEMP TABLE crash_timeline AS
(
    SELECT
        *
    FROM
        crashes
    WHERE
        EXTRACT(YEAR FROM crash_date) between '2017.0' AND '2022.0'
);

SELECT
*
FROM crash_timeline;
```

— **What are the different types of lighting conditions and the number of crashes?**

```
SELECT DISTINCT(lighting_condition) as types_of_lighting_conditions,
COUNT(*) as number_of_crashes
FROM crash_timeline
GROUP BY types_of_lighting_conditions
ORDER BY types_of_lighting_conditions;
```

Query

Query History

```
1  -- 4 What are the different types of lighting conditions and the number of crashes?
2  SELECT DISTINCT(lightning_condition) as types_of_lighting_conditions, COUNT(*) as number_of_crashes
3  FROM crash_timeline
4  GROUP BY types_of_lighting_conditions
5  ORDER BY types_of_lighting_conditions;
```

Data Output

Messages

Notifications

	types_of_lighting_conditions character varying (50)	number_of_crashes bigint
1	darkness	29810
2	darkness, lighted road	140541
3	dawn	10681
4	daylight	405170
5	dusk	18249
6	unknown	25183

–What are the different kinds of road conditions and the number of crashes?

```

SELECT DISTINCT(roadway_surface_condition) as types_of_road_conditions,
COUNT(*) as number_of_crashes
FROM crash_timeline
GROUP BY types_of_road_conditions
ORDER BY types_of_road_conditions;

```

Query

Query History

```
1  -- 5 What are the different kinds of road conditions and the number of crashes?
2  SELECT DISTINCT(roadway_surface_condition) as types_of_road_conditions, COUNT(*) as number_of_crashes
3  FROM crash_timeline
4  GROUP BY types_of_road_conditions
5  ORDER BY types_of_road_conditions;
```

Data Output

Messages

Notifications

	types_of_road_conditions character varying (25)	number_of_crashes bigint
1	dry	468910
2	ice	4553
3	other	1594
4	sand, mud, dirt	250
5	snow or slush	23511
6	unknown	46949
7	wet	83867

-- What is the crash_type and max amount difference between crash date and the date it was reported?

```
--SELECT *, (Var1 - Var2) AS Diff FROM Sample;
```

```

SELECT first_crash_type as types_of_crashes, crash_date - date_police_notified
as date_diff
From crash_timeline
ORDER BY date_diff
LIMIT 10;

```

Query		Query History
<pre> 1 -- 6 What is the crash_type and max amount difference between crash date and the date it was reported? 2 --SELECT *, (Var1 - Var2) AS Diff FROM Sample; 3 SELECT first_crash_type as types_of_crashes, crash_date - date_police_notified as date_diff 4 From crash_timeline 5 ORDER BY date_diff 6 LIMIT 10; </pre>		
Data Output		Messages
Data Output		Notifications
	types_of_crashes character varying (100)	date_diff interval
1	pedestrian	-731 days -15:31:00
2	rear end	-730 days -21:45:00
3	rear end	-730 days
4	fixed object	-702 days -00:06:00
5	pedestrian	-611 days -20:20:00
6	fixed object	-505 days -06:15:00
7	fixed object	-498 days -08:37:00
8	parked motor vehicle	-454 days -13:04:00
9	rear end	-430 days
10	parked motor vehicle	-411 days -16:25:00

-- What are the top 5 Crash Types?

-- CTE

```

WITH crash_type AS(
    -- inner query
    SELECT
        first_crash_type,
        count(*) AS cnt_crash,
        RANK() OVER (ORDER BY count(*) DESC) AS crash_rank
    FROM crash_timeline
    GROUP BY first_crash_type
)
-- outer query
SELECT
    first_crash_type as crash_type, cnt_crash
FROM crash_type
WHERE crash_rank <= 5;

```

Query Query History

```
1  -- 8 What are the top 5 Crash Types?
2  -- CTE
3  WITH crash_type AS(
4      -- inner query
5      SELECT
6          first_crash_type,
7          count(*) AS cnt_crash,
8          RANK() OVER (ORDER BY count(*) DESC) AS crash_rank
9      FROM crash_timeline
10     GROUP BY first_crash_type
11 )
12     -- outer query
13 SELECT
14     first_crash_type as crash_type, cnt_crash
15 FROM crash_type
16 WHERE crash_rank <= 5;
```

Data Output Messages Notifications

	crash_type character varying (100) 🔒	cnt_crash bigint 🔒
1	parked motor vehicle	147481
2	rear end	140314
3	sideswipe same direction	93799
4	turning	89882
5	angle	68282

-- How many road defects caused crashes?

-- CTE

```
WITH road_defects AS(
    SELECT road_defect,
           COUNT(*) AS cnt_road_df
    FROM crash_timeline
    GROUP BY road_defect
)
SELECT road_defect AS road_df, cnt_road_df AS total_road_defect
```

```
FROM road_defects
ORDER BY total_road_defect DESC;
```

```
1  -- 10 How many road defects caused crashes?
2  -- CTE
3  WITH road_defects AS(
4      SELECT road_defect,
5             COUNT(*) AS cnt_road_df
6      FROM crash_timeline
7      GROUP BY road_defect
8  )
9  SELECT road_defect AS road_df, cnt_road_df AS total_road_defect
10 FROM road_defects
11 ORDER BY total_road_defect DESC;
```

Data Output Messages Notifications

	road_df character varying (50)	total_road_defect bigint
1	no defects	515811
2	unknown	101142
3	rut, holes	4960
4	other	3488
5	worn surface	2511
6	shoulder defect	1229

-- What are the top 10 deadliest streets?

-- CTE

```
WITH deadliest_streets AS (
    SELECT
        street_name,
        injuries_fatal,
        count(*) AS cnt_injuries_fatal,
        RANK() OVER (ORDER BY count(*) DESC) AS dead_street_rank
    FROM
        crash_timeline
    WHERE injuries_fatal <> '0'
    GROUP BY
        injuries_fatal, street_name
)
SELECT
```

```
street_name, injuries_fatal, cnt_injuries_fatal
FROM deadliest_streets
WHERE dead_street_rank <=10
```

Query Query history

```
1  -- 12 What are the top 10 deadliest streets?
2  -- CTE
3  WITH deadliest_streets AS (
4      SELECT
5          street_name,
6          injuries_fatal,
7          count(*) AS cnt_injuries_fatal,
8          RANK() OVER (ORDER BY count(*) DESC) AS dead_street_rank
9      FROM
10         crash_timeline
11     WHERE injuries_fatal <> '0'
12     GROUP BY
13         injuries_fatal, street_name
14 )
15 SELECT
16     street_name, injuries_fatal, cnt_injuries_fatal
17 FROM deadliest_streets
18 WHERE dead_street_rank <=10
```

Data Output Messages Notifications			
	street_name character varying (50)	injuries_fatal character varying (2)	cnt_injuries_fatal bigint
1	Ashland Ave	1	24
2	Cicero Ave	1	22
3	Western Ave	1	20
4	Halsted St	1	19
5	Pulaski Rd	1	19
6	Archer Ave	1	13
7	Cottage Grove Ave	1	13
8	Lake Shore Dr Nb	1	13
9	Kedzie Ave	1	13
10	Lake Shore Dr Sb	1	12