**Goal 1:** Find the number of taxi rides for each taxi company for November 15-16, 2017. Sort the results by amount of trips in descending order.

**Code:**

SELECT

cabs.company\_name AS company\_name,

COUNT(trips.trip\_id) AS trips\_amount

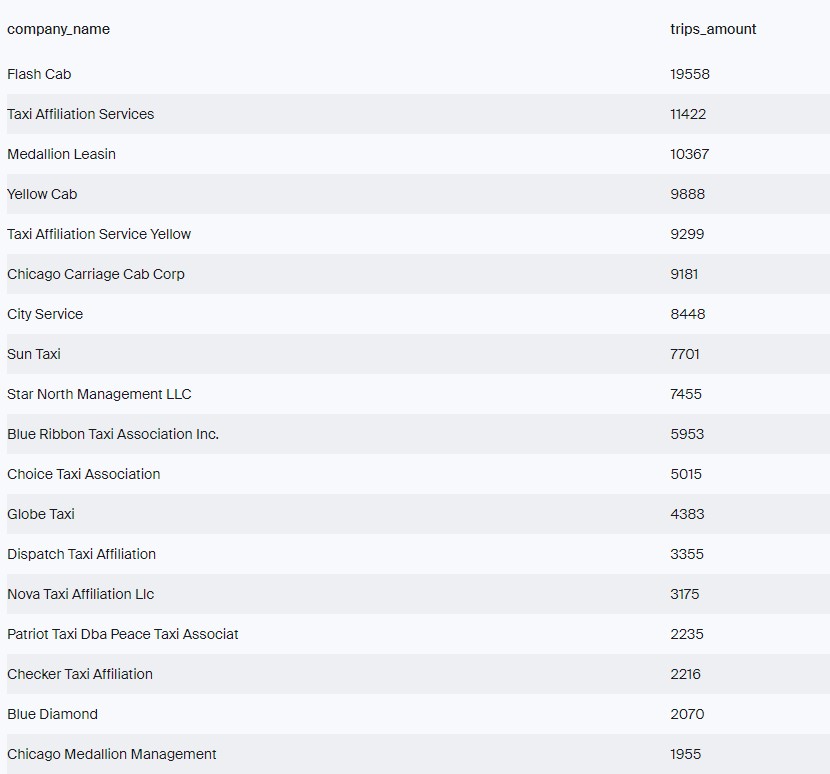
FROM cabs

INNER JOIN trips ON trips.cab\_id = cabs.cab\_id WHERE

trips.start\_ts::date BETWEEN '2017-11-15' AND '2017-11-16' GROUP BY company\_name ORDER BY

trips\_amount DESC;

**Results:**



**Goal 2:** Find the number of rides for every taxi company whose name contains the words "Yellow" or "Blue" for November 1-7, 2017. Group the results by company name.

**Code:**

SELECT

cabs.company\_name AS company\_name,

COUNT(trips.trip\_id) AS trips\_amount FROM cabs

INNER JOIN trips ON trips.cab\_id = cabs.cab\_id

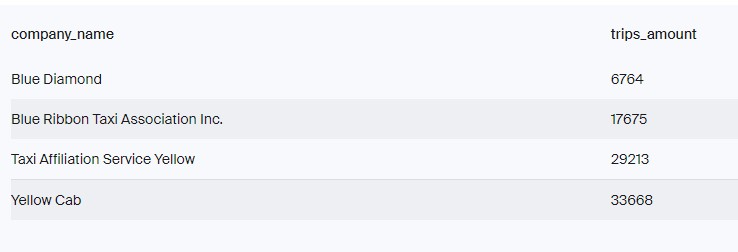
WHERE

(cabs.company\_name LIKE '%Yellow%'

OR cabs.company\_name LIKE '%Blue%')

AND trips.start\_ts::date BETWEEN '2017-11-01' AND '2017-11-07' GROUP BY company\_name;

**Results:**



**Goal 3:** Find the number of rides for the two most popular companies. Join the rides for all other companies in the group "Other." Group the data by taxi company names. Sort the result in descending order by the amount of trips.

**Code:**

SELECT

CASE

WHEN cabs.company\_name = 'Flash Cab' THEN 'Flash Cab'

WHEN cabs.company\_name = 'Taxi Affiliation Services' THEN

'Taxi Affiliation Services'

ELSE 'Other'

END AS company,

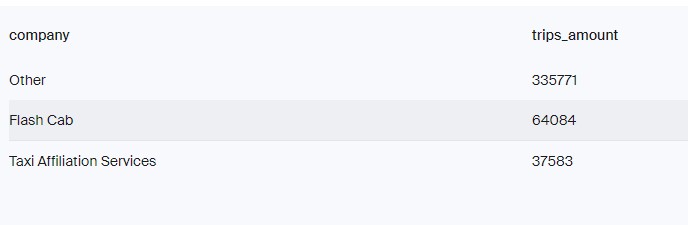
COUNT(trips.trip\_id) AS trips\_amount FROM cabs

INNER JOIN trips ON trips.cab\_id = cabs.cab\_id WHERE

trips.start\_ts::date BETWEEN '2017-11-01' AND '2017-11-07' GROUP BY company ORDER BY

trips\_amount DESC;

**Results:**



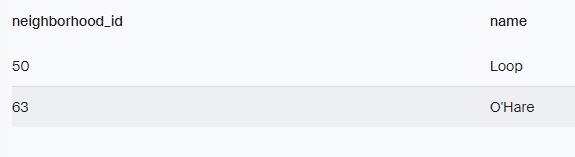
**Goal 4:** Retrieve the identifiers of the O'Hare and Loop neighborhoods.

**Code:**

SELECT neighborhood\_id, name FROM neighborhoods WHERE name LIKE 'Loop'

OR name LIKE '%Hare';

**Results:**



**Goal 5:** For each hour, retrieve the weather condition records. Break all hours into two groups: Bad if the description field contains the words rain or storm, and Good for others.

**Code:**

SELECT

DATE\_TRUNC('hour', ts::timestamp) AS date\_and\_hour,

CASE

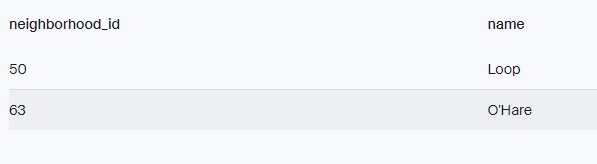
WHEN description LIKE '%rain%' THEN 'Bad'

WHEN description LIKE '%storm%' THEN 'Bad'

ELSE 'Good'

END AS weather\_conditions FROM weather\_records GROUP BY date\_and\_hour, weather\_conditions ORDER BY date\_and\_hour;

**Results:**



**Goal 6:** Retrieve all the rides that started in the Loop on a Saturday and ended at O'Hare. Get the weather conditions for each ride. Also, retrieve the duration of each ride. Ignore rides for which data on weather conditions is not available. Sort by trip id.

**Code:**

SELECT

trips.start\_ts AS start\_ts,

CASE

WHEN weather\_records.description LIKE '%rain%' THEN 'Bad'

WHEN weather\_records.description LIKE '%storm%' THEN 'Bad'

ELSE 'Good'

END AS weather\_conditions,

trips.duration\_seconds AS duration\_seconds FROM trips

INNER JOIN weather\_records ON weather\_records.ts = trips.start\_ts WHERE

trips.pickup\_location\_id = '50' AND trips.dropoff\_location\_id = '63'

AND EXTRACT(DOW FROM trips.start\_ts) = 6 ORDER BY trips.trip\_id;

**Results:**

