

1. Imprime el campo *company_name*. Encuentra la cantidad de viajes en taxi para cada compañía de taxis para el 15 y 16 de noviembre de 2017, asigna al campo resultante el nombre *trips_amount* e imprímelo también. Ordena los resultados por el campo *trips_amount* en orden descendente.

```
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
    CAST(trips.start_ts AS date) BETWEEN '2017-11-15' AND '2017-11-16'
GROUP BY
    company_name
ORDER BY
    trips_amount DESC;
```

2. Encuentra la cantidad de viajes para cada empresa de taxis cuyo nombre contenga las palabras "Yellow" o "Blue" del 1 al 7 de noviembre de 2017. Nombra la variable resultante *trips_amount*. Agrupa los resultados por el campo *company_name*.

```
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
    CAST(trips.start_ts AS date) BETWEEN '2017-11-01' AND '2017-11-07'
```

```

        AND cabs.company_name LIKE '%%Yellow%%'
GROUP BY company_name
UNION ALL
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
    CAST(trips.start_ts AS date) BETWEEN '2017-11-01' AND '2017-11-07'
    AND cabs.company_name LIKE '%%Blue%%'
GROUP BY
    company_name;

```

3. Del 1 al 7 de noviembre de 2017, las empresas de taxis más populares fueron Flash Cab y Taxi Affiliation Services. Encuentra el número de viajes de estas dos empresas y asigna a la variable resultante el nombre *trips_amount*. Junta los viajes de todas las demás empresas en el grupo "Other". Agrupa los datos por nombres de empresas de taxis. Asigna el nombre *company* al campo con nombres de empresas de taxis. Ordena el resultado en orden descendente por *trips_amount*.

```

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
    CAST(trips.start_ts AS date) BETWEEN '2017-11-01' AND '2017-11-07'
GROUP BY company
ORDER BY trips_amount DESC;

```

4. Recupera los identificadores de los barrios de O'Hare y Loop de la tabla *neighborhoods*.

```

SELECT
    neighborhood_id,
    name
FROM
    neighborhoods
WHERE
    name LIKE '%Hare' OR name LIKE 'Loop';

```

5. Para cada hora recupera los registros de condiciones meteorológicas de la tabla *weather_records*. Usando el operador CASE, divide todas las horas en dos grupos: **Bad** si el campo *description* contiene las palabras **rain** o **storm**, y **Good** para los demás. Nombra el campo resultante *weather_conditions*. La tabla final debe incluir dos campos: fecha y hora (*ts*) y *weather_conditions*.

```

SELECT
    ts,
    CASE
        WHEN description LIKE '%%rain%%' OR description LIKE '%%storm%%'
    THEN 'Bad'
        ELSE 'Good'

```

```
END AS weather_conditions
FROM
    weather_records;
```

6. Recupera de la tabla de *trips* todos los viajes que comenzaron en el Loop (*pickup_location_id*: 50) el sábado y terminaron en O'Hare (*dropoff_location_id*: 63). Obtén las condiciones climáticas para cada viaje. Utiliza el método que aplicaste en la tarea anterior. Recupera también la duración de cada viaje. Ignora los viajes para los que no hay datos disponibles sobre las condiciones climáticas.

Las columnas de la tabla deben estar en el siguiente orden:

```
start_ts
weather_conditions
duration_seconds
```

Ordena por *trip_id*.

```
SELECT
    start_ts,
    T.weather_conditions,
    duration_seconds
FROM
    trips
    INNER JOIN (
        SELECT
            ts,
            CASE
                WHEN description LIKE '%%rain%%' OR description LIKE '%%storm%%' THEN 'Bad'
                ELSE 'Good'
            END AS weather_conditions
        FROM
            weather_records
        ) T ON T.ts = trips.start_ts
WHERE
    pickup_location_id = 50 AND dropoff_location_id = 63 AND EXTRACT(DOW FROM
trips.start_ts) = 6
ORDER BY
    trips.trip_id;
```