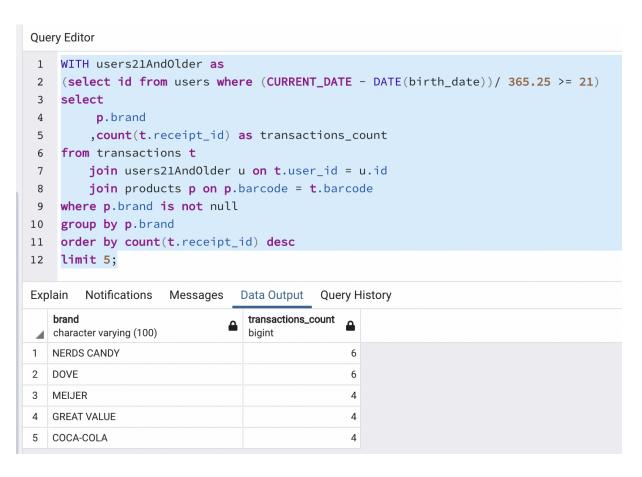
## Close Ended Questions:

### 1. What are the top 5 brands by receipts scanned among users 21 and over?

Ans. Here I'm assuming by receipts scanned means total number of receipts scanned per brand. I'm calculating the brand and their respective receipts scanned for users who are or above 21 years old. Also added a where condition to discard null brands.



# 2. What are the top 5 brands by sales among users that have had their account for at least 6 months?

Ans.

```
Query Editor
14
15
16
    WITH usersActiveFor6orMoreMonths as
    (select id from users where (CURRENT_DATE-DATE(created_date))>180 )
17
    select p.brand
18
19
             ,sum(t.sale) as total_sales
    from transactions t
20
         join usersActiveFor6orMoreMonths u on u.id = t.user_id
21
22
         join products p on p.barcode = t.barcode
    where p.brand is not null
23
    group by p.brand
24
    order by sum(t.sale) desc
25
26
    limit 5;
27
20
Explain
        Notifications
                     Messages
                                 Data Output
                                             Query History
   brand
                                  total_sales
  character varying (100)
                                  numeric
   CVS
                                       72.0000
1
2 TRIDENT
                                       46.7200
3
   DOVE
                                       42.8800
4
   COORS LIGHT
                                       34.9600
   QUAKER
                                       16.6000
5
```

### Open Ended Question:

### 2. Leading brand in the Dips and Salsa category?

Ans. I'm defining a leading brand as the brand which has a higher number of transactions and higher sales as well. I tried to find if there is a brand with less number of transactions but higher sales and there was not. In both cases there is one clear winner that is 'Tostitos'.

```
select distinct p.brand
,count(t.receipt_id) as transaction_count
,sum(t.sale) as total_sales
from products p
join transactions t on p.barcode = t.barcode
where p.category_2 = 'Dips & Salsa'
group by p.brand
order by sum(t.sale) desc;
```

```
30
     select distinct p.brand
31
32
               ,count(t.receipt_id) as transaction_count
33
               ,sum(t.sale) as total_sales
    from products p
34
     join transactions t on p.barcode = t.barcode
35
     where p.category_2 = 'Dips & Salsa'
36
     group by p.brand
37
38
     order by sum(t.sale) desc;
39
40
41
Explain
         Notifications
                       Messages
                                   Data Output
                                                 Query History
     brand
                             transaction_count
                                                total_sales
    character varying (100)
                             bigint
                                                numeric
    TOSTITOS
                                            72
                                                    260.9900
1
2
                                            42
                                                    154.3700
    [null]
3
    GOOD FOODS
                                            18
                                                    118.8900
                                            48
                                                    118.5800
4
    PACE
5
    MARKETSIDE
                                            32
                                                    103.2900
6
    FRITOS
                                            38
                                                     91.7300
7
    HELUVA GOOD!
                                            30
                                                     85.4600
8
    FRESHNESS GUARANTEED
                                            24
                                                     73.3700
```

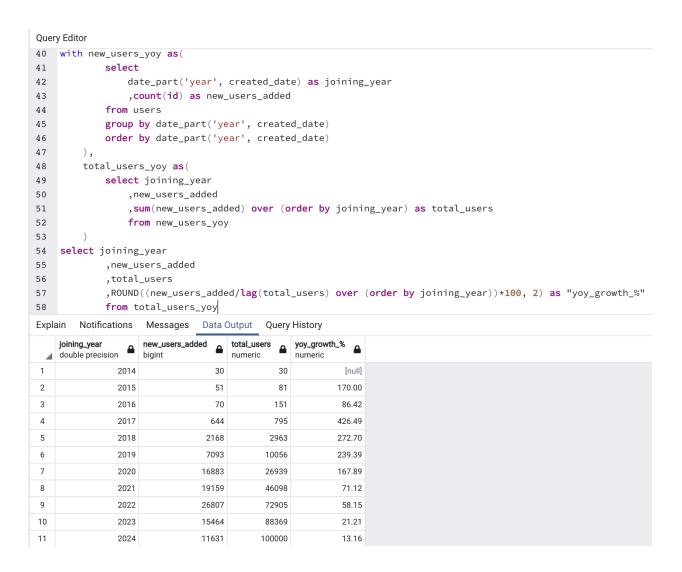
3. Year over year growth for Fetch.

Ans. There is very limited data available to find the year over year growth for Fetch. For this question, I'll be only looking at the number of customers Fetch has gained year over year. I'm using the users table and assuming all these users are active users, since there is no current way to figure out the exact active users since the transactions table also only has 3 months of data.

First I'm finding the new years in each year and also the total users using the cumulative sum. The formula I'm using to find YoY growth is (this year new users count / Last year total users count)\* 100.

```
Query:
with new_users_yoy as(
              select
                     date part('year', created date) as joining year
                     ,count(id) as new_users_added
              from users
              group by date part('year', created date)
              order by date_part('year', created_date)
       total users yoy as(
              select joining_year
                     ,new users added
                     ,sum(new users added) over (order by joining year) as total users
                     from new_users_yoy
       )
select joining_year
              ,new users added
              total users,
              ,ROUND((new_users_added/lag(total_users) over (order by joining_year))*100,
2) as "yoy_growth_%"
              from total users yoy
```

Output:



The next step is simple to calculate the average YoY growth, the output as follows. **Based on assumptions I made and looking at only the new customer base acquired each year, it is safe to say Fetch grew 152.65% YoY over the last 10 years.** 

```
Query Editor
42 with new_users_yoy as(
43
                                           select
44
                                                        date_part('year', created_date) as joining_year
45
                                                         , count(id) as new_users_added
46
                                           from users
                                             group by date_part('year', created_date)
47
48
                                           order by date_part('year', created_date)
49
                         ),
50
                            total_users_yoy as(
51
                                           select joining_year
52
                                                         ,new_users_added
                                                           , \\ \textbf{sum}(\texttt{new\_users\_added}) \ \ \textbf{over} \ \ (\textbf{order by} \ \ \texttt{joining\_year}) \ \ \textbf{as} \ \ \texttt{total\_users}
53
                                                          from new_users_yoy
54
55
                         ),
56
                         yoy_growth as
57
                            (select joining_year
58
                                         ,new_users_added
59
                                           ,total_users
                                             , \verb"ROUND" ((new_users_added/lag(total_users) over (order by joining_year)) * 100, 2) as "yoy_growth_{``}" and "yoy_growth_{``}" and "yoy_growth_{``}" are the properties of the properties of
60
61
                                            from total_users_yoy)
                             \textbf{select} \  \, \mathsf{round}( \textbf{avg}( \texttt{"yoy\_growth\_\%"}) \,, \, \, \textbf{2}) \  \, \mathsf{as} \  \, \mathsf{avg\_YoY\_growth\_percent}
62
63
                             from yoy_growth
                             where "yoy_growth_%" is not null
64
65
66
Explain Notifications Messages Data Output Query History
          avg_yoy_growth_percent_
 1
                                                       152.65
```