

- 1) Find the name of persons that made more than 5 transactions

```
with top_buyers(account_id, number_of_purchases) as
(select account_id, count(*) as number_of_purchases
from fact_transactions
join dim_accounts
using (account_id)
group by account_id
having count(*) > 5)
select name , number_of_purchases
from dim_accounts
join top_buyers
using (account_id);
```

| name | number_of_purchases |
|----------------------|---------------------|
| Erin Castro | 7 |
| Jonathan Gordon | 7 |
| Richard Russell | 6 |
| Dr. Matthew King Jr. | 7 |
| Kathryn Mendez | 6 |
| Susan Lane | 6 |
| John Gallegos PhD | 6 |
| Kara Snow | 6 |
| Justin Buck | 6 |
| Edward Trevino | 6 |
| Peter Garcia | 6 |
| Cheryl Lopez | 6 |
| Eric Gonzalez | 6 |
| Joseph Hopkins | 6 |

- 2) Find the number of positive and negative balances after all transactions are completed
(Be careful: there are accounts that have not had any transactions)

```
with total_amount_per_person(account_id , total_amount) as
(select account_id, sum(amount)
from fact_transactions
group by account_id)
select
SUM(case when dm.balance >= ta.total_amount or (dm.balance >= 0 and
ta.total_amount is null) then 1 else 0 end) as
number_of_positive_balances,
SUM(case when dm.balance <= ta.total_amount or (dm.balance <= 0 and
ta.total_amount is null) then 1 else 0 end) as
number_of_negative_balances
from dim_accounts dm
left join total_amount_per_person ta
using (account_id);
```

| count_of_positive_balances | count_of_negative_balances |
|----------------------------|----------------------------|
| 889 | 111 |

- 3) Find the name of persons who have made transactions for three consecutive days, with only one transaction on the second day

```
with consecutive_purchase_dates (account_id, date, next_purchase_date,
subsequent_purchase_date) as
(select account_id , date,
lag(date , 1, 0) over(partition by account_id order by date desc) as
next_purchase_date,
lag(date , 2, 0) over(partition by account_id order by date desc) as
subsequent_purchase_date
from fact_transactions)
select name, date, next_purchase_date, subsequent_purchase_date
from dim_accounts
join consecutive_purchase_dates
using (account_id)
where next_purchase_date = date + INTERVAL 1 DAY
and subsequent_purchase_date = date + INTERVAL 2 DAY;
```

| name | date | next_purchase_date | subsequent_purchase_date |
|-----------------|------------|--------------------|--------------------------|
| Lisa Robinson | 2022-01-21 | 2022-01-22 | 2022-01-23 |
| Lisa Mack | 2022-01-06 | 2022-01-07 | 2022-01-08 |
| Kathleen Perez | 2022-01-01 | 2022-01-02 | 2022-01-03 |
| Michael Edwards | 2022-01-17 | 2022-01-18 | 2022-01-19 |
| Jason Smith | 2022-01-11 | 2022-01-12 | 2022-01-13 |
| Maria Bradley | 2022-01-04 | 2022-01-05 | 2022-01-06 |
| Justin Buck | 2022-01-07 | 2022-01-08 | 2022-01-09 |
| Taylor Jackson | 2022-01-19 | 2022-01-20 | 2022-01-21 |