SQL Challenge #4 DATA BANK

By Kanishka Garg

PostgreSQL

PROBLEM STATEMENT

- There is a new innovation in the financial industry called Neo-Banks: new aged digital only banks without physical branches.

- Danny thought that there should be some sort of intersection between these new age banks, cryptocurrency and the data world...so he decides to launch a new initiative - Data Bank!

- Data Bank runs just like any other digital bank - but it isn't only for banking activities, they also have the world's most secure distributed data storage platform!

- Customers are allocated cloud data storage limits which are directly linked to how much money they have in their accounts. There are a few interesting caveats that go with this business model, and this is where the Data Bank team need your help!

- The management team at Data Bank want to increase their total customer base - but also need some help tracking just how much data storage their customers will need.

- This case study is all about calculating metrics, growth and helping the business analyze their data in a smart way to better forecast and plan for their future developments!

Entity relationship diagram
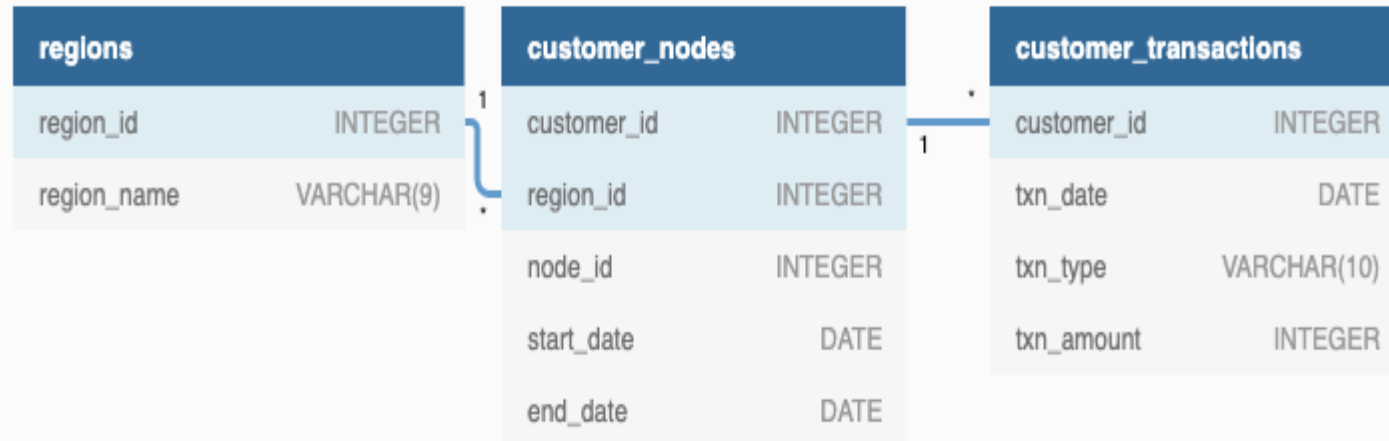
Q1: How many unique nodes are there on the Data Bank system?

```sql
select count(distinct(node_id)) as no_of_unique_nodes
from customer_nodes ;
```

no_of_unique_nodes 🔒
bigint
5

Q2: What is the number of nodes per region?

```sql
select r.region_name, count(distinct(cn.node_id)) as no_of_unique_nodes from customer_nodes cn
natural join regions r
group by r.region_name;
```

region_name character varying (9) 🔒	no_of_unique_nodes bigint 🔒
Africa	5
America	5
Asia	5
Australia	5
Europe	5

Q3: How many customers are allocated to each region?

```sql
select r.region_name, count(distinct(cn.customer_id)) as no_of_customers from customer_nodes cn
natural join regions r
group by r.region_name;
```

	region_name character varying (9) 🔒	no_of_customers bigint 🔒
1	Africa	102
2	America	105
3	Asia	95
4	Australia	110
5	Europe	88

Q4: How many days on average are customers reallocated to a different node?

```
select min(start_date), max(start_date), min(end_date), max(end_date) from customer_nodes
```

min date	max date	min date	max date
2020-01-01	2020-07-03	2020-01-02	9999-12-31

```
select round(avg(end_date - start_date)) as avg_rellocation_days from customer_nodes
where end_date != '9999-12-31'
```

avg_rellocation_days numeric
15

First, I need to know that if there is any outlier in the dataset or not. Hence, I used first query to know about max and min for each column. Then I got to know that there is one outlier in the set i.e. max of end date with year 9999. Hence, I used filter option while computing average allocation days.

Q5: What is the median, 80th and 95th percentile for this same reallocation days metric for each region?

```sql
WITH dayscte AS (
    SELECT
        cn.region_id,
        r.region_name,
        end_date - start_date AS reallocation_days
    from customer_nodes cn
    natural join regions r
    WHERE end_date != '9999-12-31')


SELECT region_id, region_name,
    PERCENTILE_CONT(0.5) WITHIN GROUP (ORDER BY reallocation_days) as Median,
    PERCENTILE_DISC(0.8) WITHIN GROUP (ORDER BY reallocation_days) as percentile_80,
    PERCENTILE_DISC(0.95) WITHIN GROUP (ORDER BY reallocation_days) as percentile_95
    FROM dayscte
    GROUP BY region_id, region_name
```

	region_id integer	region_name character varying (9)	median double precision	percentile_80 integer	percentile_95 integer
1	1	Australia	15	23	28
2	2	America	15	23	28
3	3	Africa	15	24	28
4	4	Asia	15	23	28
5	5	Europe	15	24	28

Customer Transactions

Q1: What is the unique count and total amount for each transaction type?

```sql
select txn_type, count(txn_type) as count_of_txn, to_char(sum(txn_amount), 'FM$ 999,999,999') as total_amount
from customer_transactions
group by txn_type
```

	txn_type character varying (10) 🔒	count_of_txn bigint 🔒	total_amount text 🔒
1	purchase	1617	$ 806,537
2	withdrawal	1580	$ 793,003
3	deposit	2671	$ 1,359,168

Q2: What is the average total historical deposit counts and amounts for all customers?

```
with tab1 as (select count(txn_type) as count_txn, avg(txn_amount) as total_amount
from customer_transactions
group by customer_id, txn_type
having txn_type = 'deposit')

select round(avg(count_txn)) as avg_count_txn, to_char(round(avg(total_amount), 2), 'FM$ 999,999,999') as avg_total_amount
from tab1
```

avg_count_txn numeric 🔒	avg_total_amount text 🔒
5	$ 509

Q3: or each month - how many Data Bank customers make more than 1 deposit and either 1 purchase or 1 withdrawal in a single month?

```sql
with tab2 as (SELECT customer_id, EXTRACT(MONTH FROM txn_date) as month_no,
        count(case when txn_type = 'deposit' then 1 end) as deposit,
        count(case when txn_type = 'purchase' then 1 end) as purchase,
        count(case when txn_type = 'withdrawal' then 1 end) as withdrawal
        from customer_transactions
    group by customer_id, month_no
    order by customer_id
    )

select month_no, count(distinct(customer_id)) as customer_counts from tab2
where deposit > 1 and (purchase = 1 or withdrawal = 1)
group by month_no
```

	month_no numeric 🔒	customer_counts bigint 🔒
1	1	115
2	2	108
3	3	113
4	4	50

Q4: What is the closing balance for each customer at the end of the month?

```sql
with monthly_balance_cte as (
    select customer_id, EXTRACT(MONTH FROM txn_date) as month_no,
    to_char(txn_date, 'month') as month_name,
    sum(case when txn_type = 'deposit' then txn_amount else -txn_amount end) as cls_bal
    from customer_transactions
    group by customer_id, month_no, month_name),

closing_bal_cte as (
    select customer_id, month_no, month_name,
    sum(cls_bal) over (partition by customer_id ORDER BY month_name) as runn_bal
    from monthly_balance_cte
    group by customer_id, month_no, month_name, cls_bal)

select * from closing_bal_cte
```

Output for ques 4:

	customer_id integer	month_no numeric	month_name text	runn_bal numeric
1	1	1	january	312
2	1	3	march	-640
3	2	1	january	549
4	2	3	march	610
5	3	4	april	493
6	3	2	february	-472
7	3	1	january	-328
8	3	3	march	-729
9	4	1	january	848
10	4	3	march	655
11	5	4	april	-490
12	5	1	january	464
13	5	3	march	-2413
14	6	2	february	-785
15	6	1	january	-52
16	6	3	march	340
17	7	4	april	90

Total rows: 1000 of 1720   Query complete 00:00:00.489

Q5: What is the percentage of customers who increase their closing balance by more than 5%?

```sql
WITH monthly_balance_cte as (
  SELECT customer_id,
         EXTRACT(MONTH FROM txn_date) AS month,
         SUM(CASE WHEN txn_type = 'deposit' THEN txn_amount ELSE -txn_amount END) AS closing_balance
  FROM customer_transactions
  GROUP BY customer_id, month),

closingbalance_gt5_cte AS (
  SELECT customer_id,
         sum(closing_balance) OVER (PARTITION BY customer_id ORDER BY month) as runn_bal
      FROM monthly_balance_cte
      group by customer_id, month,closing_balance),

percentage as (
    select distinct customer_id,
    (last_bal - first_bal)*100 / first_bal as growing_perc from
    (select customer_id,
    first_value(runn_bal) OVER (PARTITION BY customer_id ORDER BY customer_id) as first_bal,
    last_value(runn_bal) OVER (PARTITION BY customer_id ORDER BY customer_id) as last_bal from closingbalance_gt5_cte) as transactions
order by customer_id)

select
    count(customer_id)*100/(select count(distinct customer_id) from customer_transactions)  as percentage
from percentage
where growing_perc > 5;
```

percentage 🔒 bigint
44