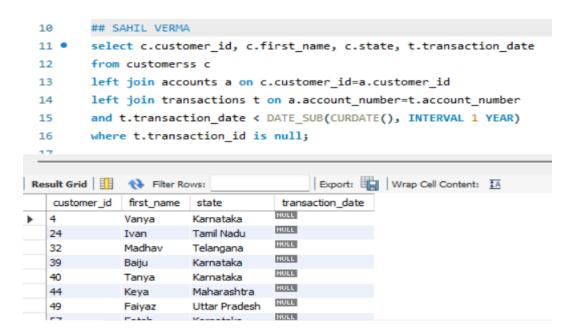
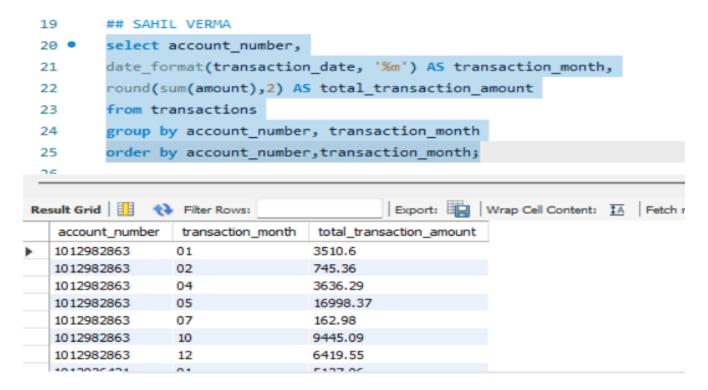
SAHIL VERMA - DS03

RE-MCT - Banking Data Analysis in SQL

1: Write a query to list all customers who haven't made any transactions in the last year. How can we make them active again? Provide appropriate region.



Q2 Summarize the total transaction amount per account per month.



Q3 Rank branches based on the total amount of deposits made in the last quarter.

```
28
         ## SAHIL VERMA
 29 • ⊖ with last quarter deposit as (
         select b.branch_id, t.transaction_type, t.transaction_date, round(sum(t.amount),2) as total_deposits
 30
         from branch b
 31
         join accounts a on b.branch id=a.branch id
 32
         join transactions t on a.account number=t.account number
 33
         group by b.branch_id, t.transaction_type, t.transaction_date
 34
         order by total deposits desc
 35
 36
         select *,
 37
         rank() over(order by total_deposits desc) as brank_ranking
 38
 39
         from last quarter deposit
         where transaction_type='Deposit' AND transaction_date <= DATE_SUB(CURDATE(), INTERVAL 3 MONTH);
 40
                                       Export: Wrap Cell Content: IA
Result Grid Filter Rows:
   branch_id transaction_type transaction_date
                                               total_deposits brank_ranking
  6
            Deposit
                            2023-08-23 11:11:28 4990.4
  25
            Deposit
                            2022-08-30 12:42:14 4987.77
            Deposit
                            2022-09-26 10:28:02 4983.37
                           2023-02-15 17:17:11 4973.02
   18
            Deposit
  20
            Deposit
                            2023-12-14 15:09:35 4968.79
  20
            Deposit
                            2024-01-30 08:07:27 4963.87
   19
            Deposit
                            2023-12-19 10:47:35 4963.68
```

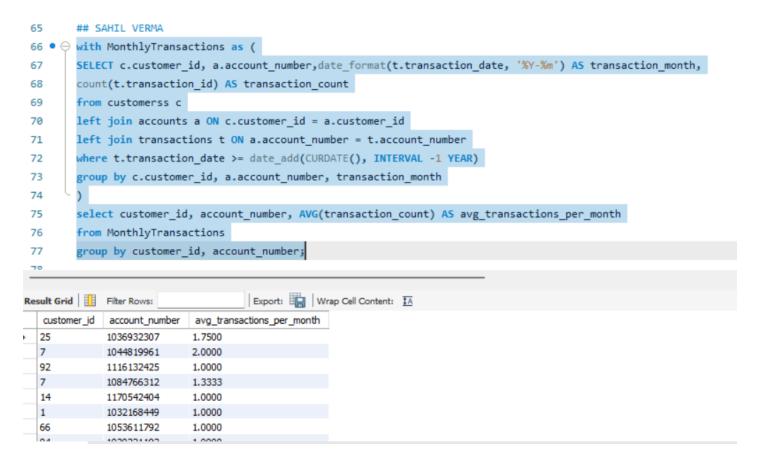
Q4 Find the name of the customer who has deposited the highest amount.

```
43
        ## SAHIL VERMA
 44 • ⊝ select * from (select c.first name, c.last name, round(sum(t.amount),2) as deposit amount
         from customerss c
 45
 46
         left join accounts a on c.customer id=a.customer id
        left join transactions t on a.account number=t.account number
 47
         where t.transaction type='Deposit'
 48
         group by c.first name, c.last name) as high deposit
 49
         order by deposit amount desc limit 1;
 50
                                          Export: Wrap Cell Content: IA
Result Grid
              ♦ Filter Rows:
   first_name
                       deposit_amount
             last_name
  Dishani
                      53763.75
            Deol
```

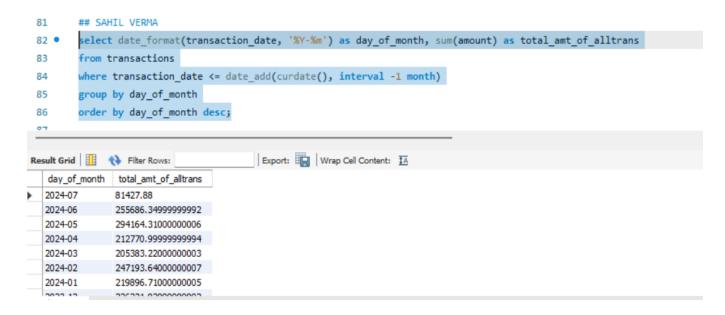
Q5 Identify any accounts that have made more than two transactions in a single day, which could indicate fraudulent activity. How can you verify any fraudulent transaction?

```
54
        ## SAHIL VERMA
 55 •
        select a.account_number, date_format(transaction_date, '%d') as day_of_month,
 56
        count(t.transaction type) as no of transaction
        from accounts a
 57
        join transactions t on a.account number=t.account number
 58
 59
        group by a.account number, day of month
        having no of transaction>3;
 60
Export: Wrap Cell Content: IA
                             no_of_transaction
                day_of_month
   account number
                             4
  1097521618
                22
  1032972218
                18
  1128643865
                03
                             4
```

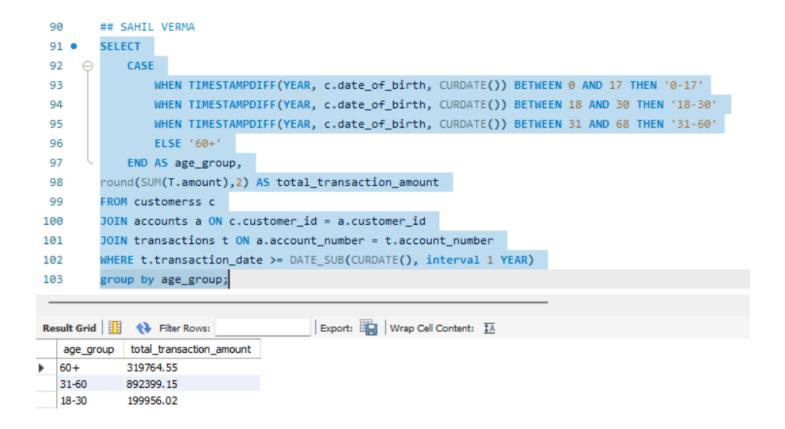
Q6 Calculate the average number of transactions per customer per account per month over the last year. average transaction per customer.



Q7 Write a query to find the daily transaction volume (total amount of all transactions) for the past month.



Q8 Calculate the total transaction amount performed by each age group in the past year. (Age groups: 0-17, 18-30, 31-60, 60+)



Q9 Find the branch with the highest average account balance.

```
## SAHIL VERMA
107
          select b.branch_name, round(avg(a.balance),2) as avg_balance
108 •
          from branch b
109
          join accounts a on b.branch_id=a.branch_id
110
111
          group by b.branch_name
112
          order by avg_balance desc;
                                               Export: Wrap Cell Content: 1A
Result Grid
               Filter Rows:
    branch_name
                           avg_balance
   ICICI 56 Lake View
                          6997.23
   BOB 101 Residency Road
                          6619.95
   HDFC 303 Banjara Hills
                          6418.17
   SBI 55 Info Park
                          6402.73
   UBI 12 Green Avenue
                          6316.28
   INDUS 123 MG Road
                          6088.83
   BOB 78 Royal Street
                          5988.37
   TOTOT FF T-F- D--I.
                          FOCO 02
Decult 170 V
```

Q10 Calculate the average balance per customer at the end of each month in last year.

