SQL Data Analyst Project

BY NANDINI WADILE

1. Detecting Recursive Fraudulent Transactions

Question:

Use a recursive CTE to identify potential money laundering chains where money is transferred from one account to another across multiple steps, with all transactions flagged as fraudulent.

Solution:

This query uses a recursive CTE to track the flow of money through multiple accounts over successive steps. The recursive part of the CTE allows us to follow the chain of transactions and identify patterns that could indicate money laundering activities. It filters out chains where all transactions are marked as fraudulent.

```
WITH  RECURSIVE fraud_chain as(
select
 nameOrig as initial_account,
 namedest as next_account,
 step,
 amount,
 newbalanceOrg
from
transactions
where isfraud = 1 and type='transfer'

union all

select fc.initial_account,
t.namedest,
t.step,
t.amount,
t.newbalanceOrg
from fraud_chain fc
join transactions t
 on fc.next_account= t.nameOrig
 and fc.step < t.step
where t.isfraud = 1
 and t.type='transfer'
 )

select*from fraud_chain;
```

2. Analyzing Fraudulent Activity over Time

Question:

Use a CTE to calculate the rolling sum of fraudulent transactions for each account over the last 5 steps.

Solution :

This query uses a CTE to calculate the cumulative sum of fraudulent transactions for each account over the last five steps. It helps in understanding the temporal distribution of fraudulent activities, which is crucial for identifying patterns over time

```
with rolling_fraud as
(select nameOrig, step,
sum(isfraud) over( partition by nameOrig order by step rows between 4 preceding and current row)
 as fraud_rolling from transactions)

select*from rolling_fraud
where fraud_rolling > 0
```

3. Complex Fraud Detection Using Multiple CTEs

Question:

Use multiple CTEs to identify accounts with suspicious activity, including large transfers, consecutive transactions without balance change, and flagged transactions.

```sql
with  large_transfers as(
select nameOrig,step,amount
from transactions
where type = 'transfer' and amount > 500000),

no_balance_change as(
select nameOrig,step,oldbalanceOrg,newbalanceOrg  from transactions
where oldbalanceOrg = newbalanceOrg ),

flagged_transactions as(
select nameOrig,step from transactions where isflaggedfraud =1)

select
  it.nameOrig
from
  large_transfers as it
join
 no_balance_change nbc on it.nameOrig= nbc.nameOrig and it.step = nbc.step
join
 flagged_transactions ft on it.nameOrig= ft.nameOrig and it.step = ft.step;
```

4. Write me a query that checks if the computed new_updated_Balance is the same as the actual newbalanceDest in the table. If they are equal, it returns those rows.

```sql
WITH CTE AS (
    SELECT
        amount,
        nameOrig,
        oldbalancedest,
        newbalancedest,
        (amount + oldbalancedest) AS new_updated_Balance
    FROM
        transactions
)
SELECT * FROM
 cte
WHERE
    new_updated_Balance = newbalancedest;
```

5. Detect Transactions with Zero Balance Before or After

Question:

Find transactions where the destination account had a zero balance before or after the transaction.

SQL Prompt:

Write a query to list transactions where oldbalanceDest or newbalanceDest is zero.

```sql
SELECT
    nameOrig,
    nameDest,
    oldbalanceDest,
    newbalanceDest,
    amount
FROM
    transactions
WHERE
    oldbalanceDest = 0 OR newbalanceDest = 0;
```

THANK YOU