

1.

sql fraud detection

```

4  -- 1. Detecting Recursive Fraudulent Transactions
5
6  WITH RECURSIVE fraud_chain as (
7      SELECT nameOrig as initial_account,
8             nameDest as next_account,
9             step,
10            amount,
11            newbalanceorig
12      FROM
13      transactions
14      WHERE isFraud = 1 and type = 'TRANSFER'
15
16      UNION ALL
17
18      SELECT fc.initial_account,
19             t.nameDest,t.step,t.amount ,t.newbalanceorig
20      FROM fraud_chain fc
21      JOIN transactions t
22      ON fc.next_account = t.nameorig and fc.step < t.step
23      where t.isfraud = 1 and t.type = 'TRANSFER')
24
25  SELECT * FROM fraud_chain;

```

Result Grid

initial_account	next_account	step	amount	newbalanceorig
C1305486145	C553264065	1	181.00	0.00
C1420196421	C972765878	1	2806.00	0.00
C137533655	C1848415041	1	20128.00	0.00
C1334405552	C431687661	1	1277212.77	0.00
C1364127192	C1136419747	1	35063.63	0.00

Result 10

2.

sql fraud detection

```

26
27  -- Analyzing fraudulent activity over time
28
29  WITH rolling_fraud AS (
30      SELECT nameOrig, step,
31             SUM(isFraud) OVER (
32                 PARTITION BY nameOrig
33                 ORDER BY step
34                 ROWS BETWEEN 4 PRECEDING AND CURRENT ROW
35             ) AS fraud_rolling
36      FROM transactions
37      LIMIT 10000
38  )
39  SELECT * FROM rolling_fraud;
40

```

Result Grid

nameOrig	step	fraud_rolling
C1000000639	249	0
C1000001337	217	0
C1000001725	46	0
C1000002591	231	0
C1000003372	167	0
C1000003615	254	0
C1000004053	327	0
C1000004530	41	0
C1000005353	226	0
C1000005555	185	0
C1000005749	380	0
C1000005789	279	0
C1000006646	306	0
C1000006873	300	0

Result 11

3.

sql fraud detection

```

42 -- 3. Complex Fraud Detection Using Multiple CTEs
43 -- Question:
44 -- Use multiple CTEs to identify accounts with suspicious activity, including large transfers, consecutive transactions without balance change, and flagged transaction
45
46 WITH large_transfers as (
47     SELECT nameOrig,step,amount FROM transactions WHERE type = 'TRANSFER' and amount >500000),
48     no_balance_change as (
49     SELECT nameOrig,step,oldbalanceOrig,newbalanceOrig FROM transactions where oldbalanceOrig=newbalanceOrig),
50     flagged_transactions as (
51     SELECT nameOrig,step FROM transactions where isflaggedfraud = 1)
52
53 SELECT
54     lt.nameOrig
55 FROM
56     large_transfers lt
57 JOIN
58     no_balance_change nbc ON lt.nameOrig = nbc.nameOrig AND lt.step = nbc.step
59 JOIN
60     flagged_transactions ft ON lt.nameOrig = ft.nameOrig AND lt.step = ft.step;
61

```

Result Grid

nameOrig
C728984460
C1100582606
C1035541766
C908544136
C689608084
C452586515
C193696150
C786455622
C19004745
C917414431

Result 12 x

Read Only

4.

sql fraud detection

```

62
63 -- 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
64
65 with CTE as (
66     SELECT amount,nameorig,oldbalancedest,newbalanceDest,(amount+oldbalancedest) as new_updated_Balance
67     FROM transactions
68 )
69 SELECT * FROM CTE where new_updated_Balance = newbalanceDest;
70
71

```

Result Grid

amount	nameorig	oldbalancedest	newbalanceDest	new_updated_Balance
84212.94	C1336775847	0.00	84212.94	84212.94
6039.61	C1893563925	31164.00	37203.61	37203.61
9104.26	C1650955365	10623.00	19727.26	19727.26
11136.72	C669725336	16180.11	27316.83	27316.83
9980.32	C1589466857	0.00	9980.32	9980.32
19772.53	C792488644	44486.99	64259.52	64259.52
8531.44	C1484275564	31817.35	40348.79	40348.79
3014.72	C1987354705	25407.65	28422.37	28422.37
139555.44	C1733619456	0.00	139555.44	139555.44
40149.38	C428245792	26426.12	66575.50	66575.50
5454.05	C470132045	18590.13	24044.18	24044.18
6522.37	C2011695710	40298.34	46820.71	46820.71
3548.72	C1596144422	13271.31	16820.03	16820.03
6159.72	C810262298	12751.13	18910.85	18910.85
3256.43	C619241052	26222.91	29479.34	29479.34
642.26	C1884422865	1772.90	2415.16	2415.16
29572.74	C62739928	12464794.41	12494367.15	12494367.15
94527.39	C1247866422	95007.35	189534.74	189534.74

Result 12 x

Read Only