**Q1: Identify how many customer’s activities are suspicious?**

**Solution:**

select count(\*)

from customer\_data

join suspicious\_activity on customer\_data.CustomerID = suspicious\_activity.CustomerID

where suspicious\_activity.SuspiciousFlag=1;

**Q1.1: Identify how many customers are fraudulent?**

**Solution:**

select count(\*)

from customer\_data

join suspicious\_activity on customer\_data.CustomerID = suspicious\_activity.CustomerID

join transaction\_records on customer\_data.CustomerID = transaction\_records.CustomerID

join fraud\_indicators on transaction\_records.TransactionID = fraud\_indicators.TransactionID

and fraud\_indicators.FraudIndicator=1;

**Q2: Identify any customers who have a history of suspicious activities but have not been previously flagged as fraudulent?**

**Solution:**

select COUNT(\*)

from customer\_data

join suspicious\_activity on customer\_data.CustomerID = suspicious\_activity.CustomerID

join transaction\_records on customer\_data.CustomerID = transaction\_records.CustomerID

join fraud\_indicators on transaction\_records.TransactionID = fraud\_indicators.TransactionID

where suspicious\_activity.SuspiciousFlag = 1

and fraud\_indicators.FraudIndicator=0;

**Q2.1: Identify any customers who have a history of suspicious activities but been previously flagged as fraudulent?**

**Solution:**

select COUNT(\*)

from customer\_data

join suspicious\_activity on customer\_data.CustomerID = suspicious\_activity.CustomerID

join transaction\_records on customer\_data.CustomerID = transaction\_records.CustomerID

join fraud\_indicators on transaction\_records.TransactionID = fraud\_indicators.TransactionID

where suspicious\_activity.SuspiciousFlag = 1

and fraud\_indicators.FraudIndicator=1;

**3-What is the percentage of fraudulent transactions based on the FraudIndicators table?**

SELECT

 COUNT(\*) AS TotalTransactions,

SUM(FraudIndicator) AS FraudulentTransactions,

  (SUM(FraudIndicator) \* 100 / COUNT(\*)) AS FraudPercentage

FROM

  Fraud\_indicators;

**Q4: Customers anomaly Score ?**

**Solution:**

select transaction\_records.CustomerID,customer\_data.Name,round(sum(anomaly\_scores.AnomalyScore),2) as Customers\_Anomaly\_Score

from anomaly\_scores

join transaction\_records on anomaly\_scores.TransactionID = transaction\_records.TransactionID

join customer\_data on transaction\_records.CustomerID = customer\_data.CustomerID

group by transaction\_records.CustomerID,customer\_data.Name

order by 1;

**Q5: Can we identify transactions with the highest anomaly score and the associated customer and merchant details?**

**Solution:**

select

customer\_data.CustomerID as CustomerID,customer\_data.Name as Customer\_Name,merchant\_data.MerchantName as Merchant\_Name,anomaly\_scores.AnomalyScore

from customer\_data

join transaction\_records on transaction\_records.CustomerID = customer\_data.CustomerID

join transaction\_metadata on transaction\_metadata.TransactionID = transaction\_records.TransactionID

join anomaly\_scores on transaction\_metadata.TransactionID = anomaly\_scores.TransactionID

join merchant\_data on transaction\_metadata.MerchantID = merchant\_data.MerchantID

GROUP BY customer\_data.CustomerID,customer\_data.Name,merchant\_data.MerchantName,anomaly\_scores.AnomalyScore

order by 4 desc;

**6- Identify the Age groups of customers who are involved in fraudulent activities.**

SELECT

CASE

        WHEN cd.age BETWEEN 0 AND 18 THEN '0-18'

        WHEN cd.age BETWEEN 18 AND 24 THEN '18-24'

        WHEN cd.age BETWEEN 25 AND 34 THEN '25-34'

        WHEN cd.age BETWEEN 35 AND 44 THEN '35-44'

        WHEN cd.age BETWEEN 45 AND 54 THEN '45-54'

        WHEN cd.age BETWEEN 55 AND 64 THEN '55-64'

        ELSE '65+'

    END AS Age\_Group,

    COUNT(DISTINCT cd.customerID) AS Number\_of\_Fraudulent\_Customers

    from customer\_data cd

    join transaction\_records tr ON cd.customerID = tr.customerID

    join transaction\_metadata tm ON tr.transactionID = tm.transactionID

    join fraud\_indicators fi ON tm.transactionID = fi.transactionID

    where fi.fraudindicator = 1

    GROUP BY Age\_groupL

    ORDER BY Age\_group;

**6.1- Is there any correlation between a customer's age and the transaction amount they make? Are older customers more likely to engage in higher value transactions?**

SELECT

CASE

        WHEN cd.age BETWEEN 0 AND 18 THEN '0-18'

        WHEN cd.age BETWEEN 18 AND 24 THEN '18-24'

        WHEN cd.age BETWEEN 25 AND 34 THEN '25-34'

        WHEN cd.age BETWEEN 35 AND 44 THEN '35-44'

        WHEN cd.age BETWEEN 45 AND 54 THEN '45-54'

        WHEN cd.age BETWEEN 55 AND 64 THEN '55-64'

        ELSE '65+'

    END AS Age\_Group,

    AVG(ad.transactionAmount) AS Avg\_Transaction\_Amount,

    sum(ad.transactionAmount) AS Total\_Transactions\_Made

FROM customer\_data cd

 JOIN  transaction\_records tr ON cd.customerID = tr.customerID

JOIN  transaction\_metadata tm ON tr.transactionID = tm.transactionID

JOIN  amount\_data ad ON tm.transactionID = ad.transactionID

 GROUP BY Age\_group

 ORDER BY Age\_group;

**6.2- identify the age group belong that belong to specific category**

SELECT tcl.Category,

       CASE

           WHEN cd.Age BETWEEN 18 AND 35 THEN 'Young'

           WHEN cd.Age BETWEEN 36 AND 55 THEN 'Adult'

           ELSE 'Old'

       END AS AgeGroup,

       COUNT(\*) AS Count

FROM transaction\_records tr

JOIN customer\_data cd

ON tr.CustomerID = cd.CustomerID

JOIN transaction\_category\_labels tcl

ON tr.TransactionID = tcl.TransactionID

GROUP BY tcl.Category, AgeGroup

ORDER BY tcl.Category, AgeGroup;

**OR**

SELECT

CASE

        WHEN cd.age BETWEEN 0 AND 18 THEN '0-18'

        WHEN cd.age BETWEEN 18 AND 24 THEN '18-24'

        WHEN cd.age BETWEEN 25 AND 34 THEN '25-34'

        WHEN cd.age BETWEEN 35 AND 44 THEN '35-44'

        WHEN cd.age BETWEEN 45 AND 54 THEN '45-54'

        WHEN cd.age BETWEEN 55 AND 64 THEN '55-64'

        ELSE '65+'

    END AS Age\_Group,

    tcl.category AS Transaction\_category,

COUNT(DISTINCT tr.transactionID) AS Number\_of\_Transactions

    from customer\_data cd

    join transaction\_records tr ON cd.customerID = tr.customerID

    join transaction\_metadata tm ON tr.transactionID = tm.transactionID

    join transaction\_category\_labels tcl ON tm.transactionID = tcl.transactionID

    GROUP BY Age\_group, transaction\_category

    ORDER BY Age\_group, number\_of\_transactions;

7**- which customer id has a higher account balance.**

SELECT CustomerID, AccountBalance

FROM account\_activity

ORDER BY AccountBalance DESC

LIMIT 1;

**8-  Which category(food or travel or retail &quot;) has done more fraud.**

SELECT

tcl.category AS categories,

  sum(ad.transactionamount) AS Fraud\_transactionsAmount

 FROM transaction\_metadata tm

 JOIN fraud\_indicators fi ON tm.TransactionID = fi.transactionID

 JOIN amount\_data ad ON tm.TransactionID = ad.transactionID

 JOIN transaction\_category\_labels tcl ON tm.TransactionID = tcl.transactionID

 WHERE fi.fraudindicator = 1

 GROUP BY categories

 ORDER BY sum(ad.transactionamount) desc;

**Q9: What is the highest transaction amount for transactions in the transaction categories ?**

**Solution:**

select 'Other' as Category,amount\_data.TransactionAmount from amount\_data

join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

where amount\_data.TransactionAmount = (select MAX(amount\_data.TransactionAmount) from amount\_data

  join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

  where transaction\_category\_labels.Category='Other')

union all

select 'Online' as Category,amount\_data.TransactionAmount from amount\_data

join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

where amount\_data.TransactionAmount = (select MAX(amount\_data.TransactionAmount) from amount\_data

  join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

  where transaction\_category\_labels.Category='Online')

union all

select 'Travel' as Category,amount\_data.TransactionAmount from amount\_data

join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

where amount\_data.TransactionAmount = (select MAX(amount\_data.TransactionAmount) from amount\_data

  join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

  where transaction\_category\_labels.Category='Travel')

union all

select 'Food' as Category,amount\_data.TransactionAmount from amount\_data

join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

where amount\_data.TransactionAmount = (select MAX(amount\_data.TransactionAmount) from amount\_data

  join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

  where transaction\_category\_labels.Category='Food')

union all

select 'Retail' as Category,amount\_data.TransactionAmount from amount\_data

join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

where amount\_data.TransactionAmount = (select MAX(amount\_data.TransactionAmount) from amount\_data

  join transaction\_category\_labels on transaction\_category\_labels.TransactionID = amount\_data.TransactionID

  where transaction\_category\_labels.Category='Retail');

**Q10: Are there specific days when fraudulent activities are more prevalent?**

**Solution:**

select a.fraud\_day,count(a.fraud\_day) as no\_of\_frauds

from

(select dayname(transaction\_metadata.Timestamp) as Fraud\_day from transaction\_metadata

join fraud\_indicators on transaction\_metadata.TransactionID = fraud\_indicators.TransactionID

where fraud\_indicators.FraudIndicator=1) a

group by a.fraud\_day;

**Q10.1: Are there specific Monts when fraudulent activities are more prevalent?**

**Solution:**

select a.fraud\_day,count(a.fraud\_day) as no\_of\_frauds

from

(select monthname(transaction\_metadata.Timestamp) as Fraud\_day from transaction\_metadata

join fraud\_indicators on transaction\_metadata.TransactionID = fraud\_indicators.TransactionID

where fraud\_indicators.FraudIndicator=1) a

group by a.fraud\_day;

**11-Are there particular transaction amounts that are more frequently associated with fraud?**

SELECT transaction\_records.TransactionID, ROUND(Amount, 1) AS RoundedAmount, FraudIndicator

FROM transaction\_records

JOIN fraud\_indicators

ON transaction\_records.TransactionID = fraud\_indicators.TransactionID

WHERE FraudIndicator = 1

ORDER BY RoundedAmount DESC;

**12.-How does the anomaly score distribution vary between different transaction categories?**

SELECT

       tcl.category AS Category,

       ROUND(AVG(sc.anomalyscore),2) AS Avg\_anomalyscore,

       ROUND(MIN(sc.anomalyscore),2) AS Min\_anomalyscore,

       ROUND(MAX(sc.anomalyscore),2) AS Max\_anomalyscore,

       ROUND(STDDEV(sc.anomalyscore),2) AS Stddev\_anomalyscore

FROM transaction\_category\_labels tcl

 JOIN transaction\_metadata tm ON tcl.transactionID = tm.transactionID

 JOIN anomaly\_scores sc ON tm.transactionID = sc.transactionID

GROUP BY tcl.category

ORDER BY tcl.category;