**SQL QUERIES**

**Metric 1 : Avg. transactions per month (Retained vs Non-Retained)**

WITH cte AS(SELECT user\_id,CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status,count(\*) as txn\_count from transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status, ROUND(avg(txn\_count),2) as avg\_transactions from cte

GROUP BY retention\_status

**Metric 2: Average Time between activation and first transaction**

WITH cte AS(SELECT user\_details.user\_id,

CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status,

MIN(date\_of\_transaction)-activation\_date as days\_to\_first\_txn

FROM transaction\_details

INNER JOIN user\_details ON transaction\_details.user\_id=user\_details.user\_id

WHERE status='Completed'

GROUP BY user\_details.user\_id)

SELECT retention\_status,AVG(days\_to\_first\_txn) as avg\_days2\_first\_txn FROM cte

GROUP BY retention\_status

**Metric 3: Transaction type distribution (Merchant vs P2P vs Card)**

WITH cte AS(SELECT user\_id,

CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status

FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status,type\_of\_transaction,count(\*) as txn\_count,

CONCAT(ROUND(100.0 \* count(\*) / SUM(COUNT(\*)) OVER (PARTITION BY retention\_status),2),'%')as percentage\_share

FROM transaction\_details

INNER JOIN cte oON transaction\_details.user\_id=cte.user\_id

WHERE status='Completed'

GROUP BY retention\_status,type\_of\_transaction

ORDER BY txn\_count desc

**Metric 4: Avg, Min, Max transaction value**

WITH cte AS(SELECT user\_id,

CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction)) >=3 then 'Retained' ELSE 'Not Retained' end as retention\_status

FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status,AVG(amount) as avg\_txn\_value,MAX(amount) as max\_txn\_value,MIN(amount) as min\_txn\_value FROM transaction\_details

INNER JOIN cte ON transaction\_details.user\_id=cte.user\_id

WHERE status='Completed'

GROUP BY retention\_status

**Metric 5: Age group distribution (Teens, Young Adults, Adults)**

WITH cte AS(SELECT user\_id,

CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status

FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status, CASE WHEN age between 13 and 17 then 'Teen(13-17)' WHEN age between 18 and 21 then 'Young Adult(18-21)' WHEN age between 22 and 25 then 'Adult(22-25)' else 'Other' end as age\_distribution,

COUNT(\*) as users from user\_details

INNER JOIN cte on user\_details.user\_id=cte.user\_id

GROUP BY retention\_status,age\_distribution

ORDER BY retention\_status, users desc

**Metric 6: Average active months(Retained vs Non Retained)**

WITH active\_months AS (SELECT user\_id,count(distinct date\_trunc('month',date\_of\_transaction)) AS active\_months FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id),

retention\_status AS (SELECT user\_id,CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status,AVG(active\_months) AS avg\_active\_months

FROM retention\_status

INNER JOIN active\_months ON retention\_status.user\_id=active\_months.user\_id

GROUP BY retention\_status

**Metric 7: Transaction type diversity (Single vs Multi-type users)**

WITH retention\_status AS (SELECT user\_id, CASE WHEN COUNT(DISTINCT date\_trunc('month',date\_of\_transaction))>=3 then 'Retained' else 'Not Retained' end as retention\_status

FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id),

txn\_type AS (SELECT user\_id,

COUNT(DISTINCT type\_of\_transaction) AS txn\_type\_count

FROM transaction\_details

WHERE status='Completed'

GROUP BY user\_id)

SELECT retention\_status, SUM(Case when txn\_type\_count>=2 then 1 else 0 end) as multiple\_users, COUNT(\*) AS total\_users,

CONCAT(ROUND(100 \* SUM(Case when txn\_type\_count>=2 then 1 else 0 end)/count(\*) ,2),'%') AS percentage\_of\_multipletype\_users

FROM retention\_status

INNER JOIN txn\_type ON retention\_status.user\_id=txn\_type.user\_id

GROUP BY retention\_status