**SQL CASE STUDY – BANKING SERVICES BY DATA IN MOTION**

--1. Find the total number of transactions each customer made in 2023.

SELECT

CONCAT(C.FIRSTNAME,' ',C.LASTNAME) AS FULL\_NAME,

COUNT(T.ACCOUNTID) AS NO\_OF\_TRANSACTIONS

FROM TRANSACTIONS T

INNER JOIN ACCOUNTS A

ON T.ACCOUNTID = A.ACCOUNTID

INNER JOIN CUSTOMERS C

ON A.CUSTOMERID = C.CUSTOMERID

WHERE DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY FULL\_NAME;

--2. Calculate the total income (positive transactions) for each customer in 2023.

SELECT

CONCAT(C.FIRSTNAME,' ',C.LASTNAME) AS FULLNAME,

SUM(AMOUNT) AS TOTAL\_INCOME

FROM TRANSACTIONS T

INNER JOIN ACCOUNTS A

ON T.ACCOUNTID = A.ACCOUNTID

INNER JOIN CUSTOMERS C

ON A.CUSTOMERID = C.CUSTOMERID

WHERE SIGN(AMOUNT) != -1 AND DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY FULLNAME

ORDER BY FULLNAME;

--3 Find the month with the highest total expenses (negative transactions) in 2023.

WITH EXPENSE AS (

SELECT

DATE\_PART('MONTH',TRANSACTIONDATE) AS MONTH

,SUM(ABS(AMOUNT)) AS TOTAL\_EXPENSES

FROM TRANSACTIONS

WHERE SIGN(AMOUNT) = -1 AND DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY MONTH

)

SELECT

\*

FROM EXPENSE

WHERE TOTAL\_EXPENSES = (SELECT

MAX(TOTAL\_EXPENSES)

FROM EXPENSE);

--4. Calculate the total income, total expenses, and net amount (income - expenses) for each account type for the year 2023.

SELECT

A.ACCOUNTTYPEID, AT.ACCOUNTTYPENAME,

SUM(CASE WHEN SIGN(T.AMOUNT) != -1 THEN T.AMOUNT END) AS INCOME,

SUM(CASE WHEN SIGN(T.AMOUNT) = -1 THEN T.AMOUNT END) AS EXPENSES,

SUM(AMOUNT) AS NET\_AMOUNT

FROM TRANSACTIONS T

INNER JOIN ACCOUNTS A

ON T.ACCOUNTID = A.ACCOUNTID

INNER JOIN ACCOUNTTYPES AT

ON A.ACCOUNTTYPEID = AT.ACCOUNTTYPEID

WHERE DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY A.ACCOUNTTYPEID,AT.ACCOUNTTYPENAME

ORDER BY A.ACCOUNTTYPEID;

--5. Find the customer with the highest average monthly net transactions (income - expenses) in 2023.

WITH AMOUNT AS(

SELECT

DATE\_PART('MONTH',TRANSACTIONDATE) AS MONTH

,A.CUSTOMERID

,SUM(AMOUNT) AS NET\_AMOUNT

FROM TRANSACTIONS T

INNER JOIN ACCOUNTS A

ON T.ACCOUNTID = A.ACCOUNTID

WHERE DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY MONTH,A.CUSTOMERID

)

--- Getting the average montly net

,AVERAGE\_AMT AS(

SELECT

CUSTOMERID,

ROUND(AVG(NET\_AMOUNT),0) AS AVG\_AMOUNT

FROM AMOUNT

GROUP BY CUSTOMERID

)

--- Getting the customer details

SELECT

DISTINCT A.CUSTOMERID

,CONCAT(C.FIRSTNAME,' ',C.LASTNAME) AS FULLNAME

FROM AVERAGE\_AMT A

INNER JOIN CUSTOMERS C

ON A.CUSTOMERID = C.CUSTOMERID

WHERE AVG\_AMOUNT = (

SELECT

MAX(AVG\_AMOUNT)

FROM AVERAGE\_AMT

);

--6. For each month in 2023, identify the customer with the highest net transactions (income - expenses) and rank them according to this value.

WITH AMOUNT AS(

SELECT

DATE\_PART('MONTH',TRANSACTIONDATE) AS MONTH

,A.CUSTOMERID

,SUM(AMOUNT) AS NET\_AMOUNT

,DENSE\_RANK() OVER(PARTITION BY DATE\_PART('MONTH',TRANSACTIONDATE) ORDER BY SUM(AMOUNT) DESC) AS DRNK

FROM TRANSACTIONS T

INNER JOIN ACCOUNTS A

ON T.ACCOUNTID = A.ACCOUNTID

WHERE DATE\_PART('YEAR',TRANSACTIONDATE) = '2023'

GROUP BY MONTH,A.CUSTOMERID

)

SELECT

MONTH

,CUSTOMERID

,CONCAT(C.FIRSTNAME,' ',C.LASTNAME) AS FULLNAME

,NET\_AMOUNT,

DENSE\_RANK() OVER(ORDER BY NET\_AMOUNT DESC) AS RANK

FROM AMOUNT A

INNER JOIN CUSTOMERS C

USING (CUSTOMERID)

WHERE DRNk=1

ORDER BY RANK