**MAZSQL01 – HOMEWORK**

**SESSION 4**

|  |
| --- |
| **Description:**  *Paytm is an Indian multinational financial technology company. It specializes in digital payment systems, e-commerce and financial services. Paytm wallet is a secure and RBI (Reserve Bank of India)-approved digital/mobile wallet that provides a myriad of financial features to fulfill every consumer’s payment needs. Paytm wallet can be topped up through UPI (Unified Payments Interface), internet banking, or credit/debit cards. Users can also transfer money from a Paytm wallet to the recipient's bank account or their own Paytm wallet.*  Below is a small database of payment transactions from 2019 to 2020 of Paytm Wallet. The database includes 6 tables:   * **fact\_transaction**: Store information of all types of transactions: Payments, Top-up, Transfers, Withdrawals * **dim\_scenario**: Detailed description of transaction types * **dim\_payment\_channel**: Detailed description of payment methods * **dim\_platform**: Detailed description of payment devices * **dim\_status**: Detailed description of the results of the transaction |

**PART 1: PRACTICE**

**Task 1: Hoàn thành câu truy vấn dưới đây với lệnh GROUP BY**

1.1 Khi chúng ta cần gom nhóm những dòng dữ liệu cùng tính chất và tính toán các lệnh **SUM, COUNT, MIN, MAX, AVG thì phải dùng GROUP BY**.

* Trong bảng dữ liệu **fact\_transaction\_2019** mỗi giao dịch thành công sẽ ứng với một loại category khác nhau. **Hãy tính toán số lượng giao dịch của từng loại** trên để thuận tiện cho việc báo cáo.
* Hãy nhớ lấy câu: “**GROUP BY cái gì thì mình SELECT cái đó**”

|  |
| --- |
| SELECT category  ,COUNT(transaction\_id) as number\_trans  FROM fact\_transaction\_2019 AS fact\_19  LEFT JOIN dim\_scenario AS dim\_scen  ON fact\_19.scenario\_id = dim\_scen.scenario\_id  LEFT JOIN dim\_status AS stat  ON fact\_19.status\_id = stat.status\_id  WHERE status\_description = 'success'  GROUP BY category |

**1.2 Thực hành truy vấn với một bảng CTE**

* **Bước 1**: Tạo một CTE lưu trữ một bảng tạm chứa thông tin về tất cả các giao dịch trong năm 2019 thoả mãn điều kiện giao dịch thành công và thanh toán bằng hình thức “Bank account”. Chọn ra những thông tin: transaction\_id, customer\_id, charged\_amount, platform\_id.
* **Bước 2**: Join với bảng dim\_platform để giữ lại các giao dịch được thực hiện bằng thiết bị “Android”

|  |
| --- |
| WITH trans\_table AS (  SELECT transaction\_id  , customer\_id  , charged\_amount  , platform\_id  FROM fact\_transaction\_2019 AS fact\_19  JOIN dim\_payment\_channel AS chan  ON fact\_19.payment\_channel\_id = chan.payment\_channel\_id  WHERE payment\_method = 'Bank account' AND status\_id = 1  )  SELECT transaction\_id  ,customer\_id  ,charged\_amount  ,payment\_platform  FROM trans\_table  JOIN dim\_platform AS plat  ON trans\_table.platform\_id = plat.platform\_id  WHERE payment\_platform = 'android' |

**1.3. Sử dụng truy vấn lồng (Subquery) và CTE.**

* Bạn hãy cho biết tổng số giao dịch của từng loại **transaction\_type**, với điều kiện: giao dịch thành công, thời gian giao dịch trong 3 tháng đầu tiên của năm 2019
* Tính tỷ trọng số lượng giao dịch của từng loại trên tổng số giao dịch trong 3 tháng.

Hướng giải quyết cơ bản của bài này sẽ như sau:

1. Mình cần tạo ra một CTE chọn lọc dữ liệu thỏa mãn yêu cầu đề bài.
2. Tạo một CTE: Gom nhóm, tính toán số lượng giao dịch của từng loại trên và tổng số giao dịch thành công của 3 tháng đầu tiên
3. Hiển thị top 5 loại giao dịch có số lượng cao nhất và tỉ lệ của mỗi loại

|  |
| --- |
| --- Bước 1: CTE thoả mãn yêu cầu thành công và giao dịch trong 3 tháng đầu tiên  WITH joined\_table AS (  SELECT fact\_19.\*     , transaction\_type  FROM fact\_transaction\_2019 AS fact\_19  LEFT JOIN dim\_status as stat     ON fact\_19.status\_id = stat.status\_id  LEFT JOIN dim\_scenario AS scen     ON fact\_19.scenario\_id = scen.scenario\_id  WHERE fact\_19.status\_id = 1     AND MONTH(transaction\_time) <= 3  )  --- Bước 2: CTE gom nhóm và tính toán theo từng loại giao dịch.  -- NOTES: Từ CTE thứ hai trở đi chúng ta sẽ sử dụng dấu ',' để ngăn cách các CTE như bên dưới  , total\_table AS (  SELECT transaction\_type     , count(transaction\_id) AS number\_trans     , (SELECT COUNT(transaction\_id) FROM joined\_table) AS total\_trans -- subquery lấy ra total trans  FROM joined\_table  GROUP BY transaction\_type  )  --Bước 3: Hiển thị vừa đủ thông tin đề yêu cầu  SELECT TOP 5 \*     , FORMAT ((number\_trans\*1.0) / (total\_trans\*1.0), 'p') as pct  FROM total\_table  ORDER BY (number\_trans\*1.0) / (total\_trans\*1.0) DESC |

**Part 2: APPLYING FOR EXERCISES**

# Task 1: Retrieve an overview report of payment types

## Paytm has a wide variety of transaction types in its business. Your manager wants to know the contribution (by percentage) of each transaction type to total transactions. Retrieve a report that includes the following information: transaction type, number of transactions and proportion of each type in total. These transactions must meet the following conditions:

* Were created in 2019
* Were paid successfully

Show only the results of the top 5 types with the highest percentage of the total.

-- Retrieve transaction type, number of transactions and proportion of each type in total.

WITH joined\_table AS (

SELECT fact\_19.\*

   , transaction\_type

FROM fact\_transaction\_2019 AS fact\_19 -- ●  Were created in 2019

LEFT JOIN dim\_status as stat

   ON fact\_19.status\_id = stat.status\_id

LEFT JOIN dim\_scenario AS scen

   ON fact\_19.scenario\_id = scen.scenario\_id

WHERE fact\_19.status\_id = 1 -- Were paid successfully

)

, total\_table AS (

SELECT transaction\_type

   , count(transaction\_id) AS number\_trans

   , (SELECT COUNT(transaction\_id) FROM joined\_table) AS total\_trans -- subquery lấy ra total trans

FROM joined\_table

GROUP BY transaction\_type

)

SELECT TOP 5 \*

   , FORMAT ((number\_trans\*1.0) / (total\_trans\*1.0), 'p') as pct

FROM total\_table

ORDER BY (number\_trans\*1.0) / (total\_trans\*1.0) DESC

## After your manager looks at the results of these top 5 types, he wants to deep dive more to gain more insights.

Retrieve a more detailed reportwith following information: **transaction type, category, number of transactions and proportion of each category in the total of that transaction type**. These transactions must meet the following conditions:

* Were created in 2019
* Were paid successful

WITH joined\_table AS (

    SELECT fact\_19.\*

        , transaction\_type

        , category

    FROM fact\_transaction\_2019 AS fact\_19 -- Were created in 2019

    LEFT JOIN dim\_status AS stat

    ON fact\_19.status\_id = stat.status\_id

    LEFT JOIN dim\_scenario AS scen

    ON fact\_19.scenario\_id = scen.scenario\_id

    WHERE fact\_19.status\_id = 1 -- Were paid successful

)

, trans\_type\_table AS (

    SELECT transaction\_type -- transaction type

        ,  count(transaction\_id) AS number\_trans\_type-- number of transactions

    FROM joined\_table

    GROUP BY transaction\_type

)

, cat\_table AS (

    SELECT transaction\_type

        , category          -- category

        , count(transaction\_id) AS number\_trans\_category -- and proportion of each category in the total of that transaction type.

    FROM joined\_table

    GROUP BY transaction\_type, category

)

SELECT trans\_type\_table.transaction\_type

    , category

    , number\_trans\_category

    , number\_trans\_type

    , format(number\_trans\_category\*1.0/number\_trans\_type\*1.0,'p') AS pct

FROM trans\_type\_table

LEFT JOIN cat\_table

ON trans\_type\_table.transaction\_type = cat\_table.transaction\_type

GROUP BY trans\_type\_table.transaction\_type, category, number\_trans\_category, number\_trans\_type

ORDER BY trans\_type\_table.transaction\_type ASC

# Task 2: Retrieve an overview report of customer’s payment behaviors ( hành vi thanh toán khách hàng )

## Paytm has acquired a lot of customers. Retrieve a report that includes the following information: the number of transactions, the number of payment scenarios, the number of payment categories and the total of charged amount of **each customer**.

* Were created in 2019
* Had status description is successful
* Had transaction type is payment

WITH joined\_table AS (

    SELECT fact\_19.\*

        , category

    FROM fact\_transaction\_2019 AS fact\_19 -- Were created in 2019

    LEFT JOIN dim\_status AS stat

    ON fact\_19.status\_id = stat.status\_id

    LEFT JOIN dim\_scenario AS scen

    ON fact\_19.scenario\_id = scen.scenario\_id

    WHERE transaction\_type = 'Payment' AND fact\_19.status\_id = 1 -- Had status description is successful + transaction type payment

)

, total\_table AS (

SELECT customer\_id

    , COUNT(transaction\_id) AS number\_trans-- the number of transactions

    , COUNT(distinct(scenario\_id)) AS number\_scenarios -- the number of payment scenarios

    , COUNT(distinct(category)) AS number\_categories

    , SUM(charged\_amount) AS total\_amount -- the total of charged amount of each customer.

FROM joined\_table

GROUP BY customer\_id

)

SELECT \*

FROM total\_table

ORDER BY number\_trans DESC