MySQL Analysis Sales Report

Information and insights from tables of data sales

* Understanding the sales dataset to get insight into products, revenue, commission, churn and sales trends is essential. Analyzing a dataset in SQL, this mini project involved two datasets.

* The scheme for commission was paid for three times to sales, the 1st commission is labelled 'ST' in the 1st month, the 2nd one is labelled 'ND' in the 4th month, the 3rd one is labelled ‘RD’ in the 7th month

* Should the customer have disconnected the service , is designated on CHURN_DATE , thus the commission would not be paid

* The commission is paid 200% of revenue, 100% is the 1st month, and in the 4th and 7th months of each, 50%

* The table detail_transaksi consists of ADD_INT_PRODUCT, AGENCY, CHURN, CITY, COMMISSION_GROSS, COMPLETED_INSTALL, CUST_ACCT, DEPARTMENT, KOMISI, NILAI_REVENUE_PRODUCT, PAYMENT_STATUS, POSITION, REPORTED_DATE, SALES_CODE, TERM_PAYMENT

* The table _transaksi was changed into table detail_2 due to incompatible format date column 'dd/mm/yyyy' , using Google Colab, the format was extracted to be 'yyyy-mm-dd'

* The table agent_sales consists of AGENCY, CITY, DEPARTMENT, PENDIDIKAN, POSITION, SC

Combining these datasets, the insight trend of sales either quarterly or monthly, and how the product, department have relationship to churn is reliable to get critical and purposeful information to drive next decision

Creating database and tables

1a.Creating table detail_transaksi

```sql
CREATE DATABASE vms;
```

```sql
CREATE TABLE detail_transaksi (
    REPORTED_DATE date,
    TERM_PAYMENT VARCHAR (10),
    KOMISI VARCHAR (10),
    DEPARTMENT VARCHAR (50),
    COMPLETED_INSTALL date,
    CUST_ACCT int (10) PRIMARY KEY,
    ADD_INT_PRODUCT VARCHAR (10),
    SALES_CODE VARCHAR (10),
    POSITION VARCHAR (20),
    AGENCY VARCHAR (10),
    CITY VARCHAR (10),
    PAYMENT_STATUS VARCHAR (10),
    NILAI_REVENUE_PRODUCT int (10),
    CHURN_DATE date,
    COMMISSION_GROSS int (10)
);
```

1b. Creating table agent_sales, CUST_ACCT as FOREIGN KEY.

```sql
CREATE TABLE vms.agent_sales (
    CUST_ACCT int (10),
    SALES_CODE VARCHAR (5),
    DEPARTMENT VARCHAR (30),
    POSITION VARCHAR (20),
    AGENCY VARCHAR (7),
    CITY VARCHAR (20),
    PENDIDIKAN VARCHAR (5),
    FOREIGN KEY (CUST_ACCT) REFERENCES vms.detail_transaksi (CUST_ACCT)
);
```

2a. Inserting data into the table detail_transaksi

```
INSERT INTO vms.detail_transaksi (
    REPORTED_DATE,TERM_PAYMENT,KOMISI,DEPARTMENT,COMPLETED_INSTALL,CUST_ACCT,ADD_INT_PRODUCT,SALES_CODE,POSITION,
    AGENCY,CITY,PAYMENT_STATUS,NILAI_REVENUE_PRODUCT,CHURN_DATE,COMMISSION_GROSS)
    VALUES ('2023-04-17','JAN','ST','SA SURABAYA','2023-03-12',467,'PD19','KLH9','Agent
vendor','AG2','SURABAYA','PAID',325500,'2023-04-16',0);
```

2b. Inserting data into the table agent_sales

```
INSERT INTO vms.agent_sales (
    CUST_ACCT,SALES_CODE,DEPARTMENT,POSITION,AGENCY,CITY,PENDIDIKAN)
    VALUES (467,'KLH9','SA SURABAYA','Agent vendor','AG2','SURABAYA','D3');
```

QUERIES TO ANALYZE

COMMISSION - NEW ACCOUNTS - REVENUE

1. Total records for each table, count unique customers & sales code for sales agent.
2. Total new installations for each department.
3. Total records: product, agency, sales agent, department, city.
4. Total new accounts, revenue and commission by city.
5. Group by city for all new accounts .
6. Group by monthly installation for paid accounts, ordered by revenue.
7. Group by quarterly commission gross, trend commission for new accounts.
8. Group by department for agent sales that have revenue > Rp.5.000.000, - ordered by revenue.
9. Group by department that has the TOP 5 new accounts and order by total new accounts.
10. Average revenue of each department.
11. The trend of quarterly paid revenue.

12. The trend of quarterly new accounts.

13. To change the date format using Google Collab, extract a new dataset into detail_2 and its visualization.

14. The trend of new monthly accounts.

15. The trend of quarterly total commission.

16. The trend of monthly total commission.

PRODUCT

17 Group by product: the total new accounts with product sold, total revenue and average sales.

18. 10 TOP products stayed ‘ACTIVE’ either for the fourth or the seventh month.

19. Trend of the quarterly most popular product.

20. Trend of the monthly most popular product.

21. Profit (total revenue for 4 months), group by product.

22. The best-selling product by department using CTE

23. The best-selling product by city.

24. The top 10 best-selling products by sales agent.

CHURN

25. Total accounts, total churned accounts, percentage of churn accounts in the second commission.

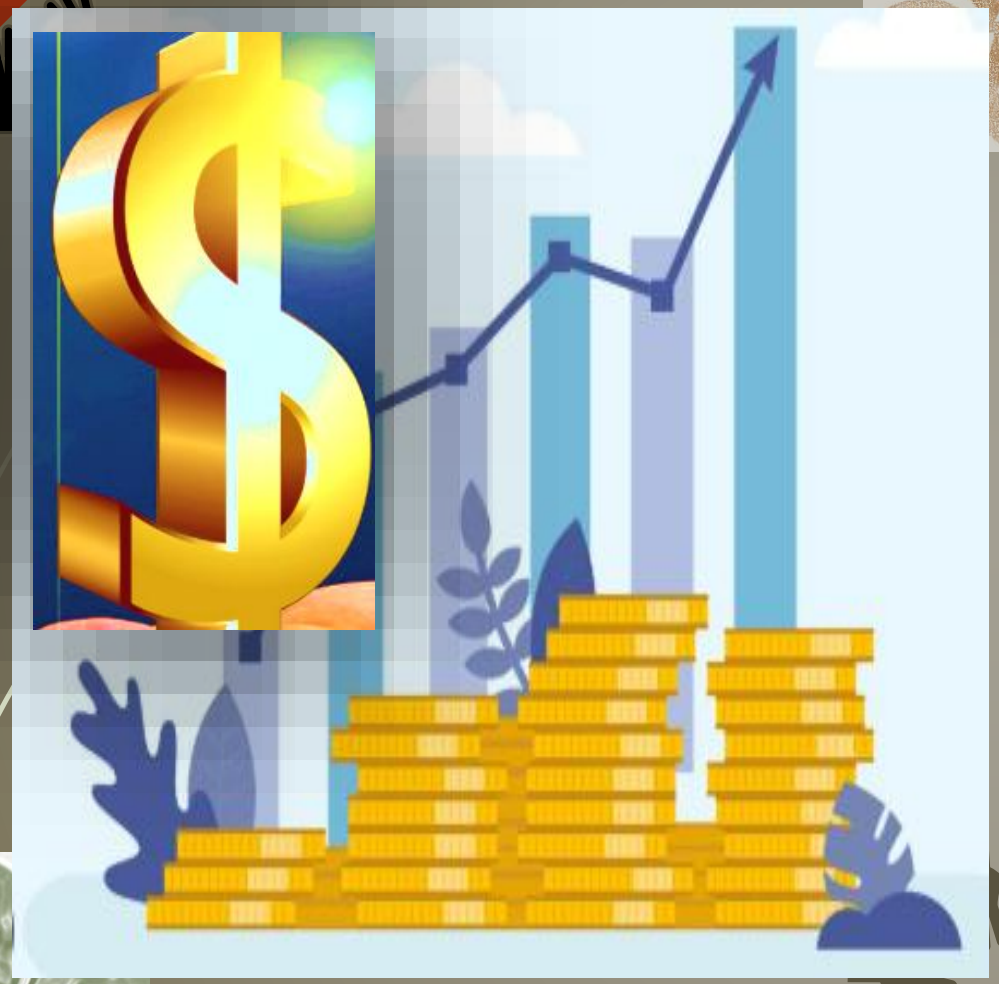
26. Total accounts, total churned accounts, percentage of churn accounts in the third commission.

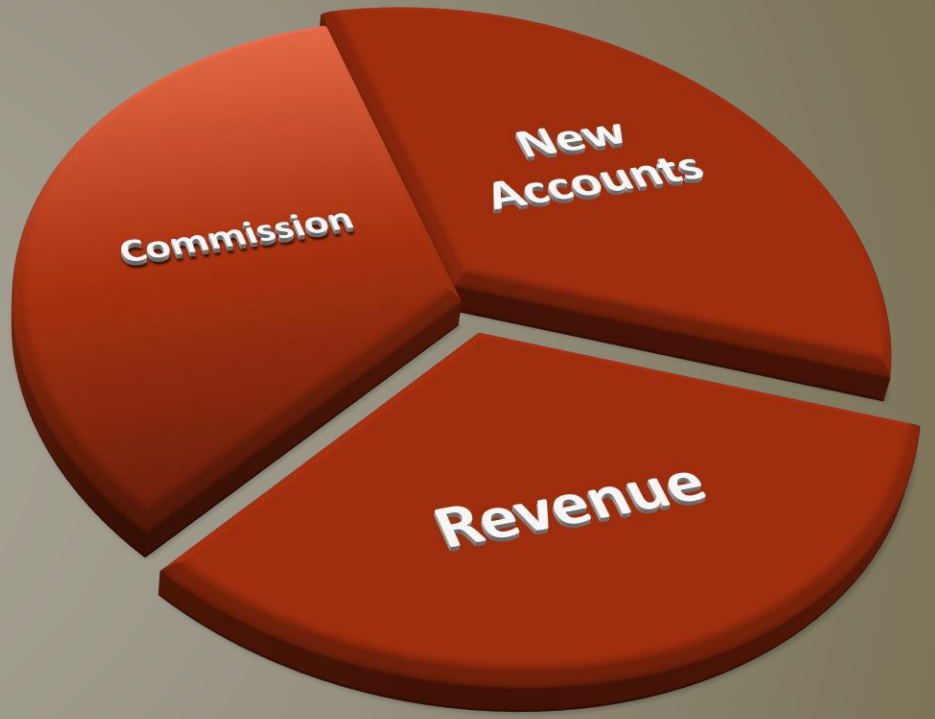
27. The top 15 most churned products, revenue, total revenue for 4 months of subscription, total churned accounts and total commission for the 7th month.

28. Trend of total churned accounts in the 4th & 7th months.

29. Group by department, the total new paid accounts, total churned accounts in the 4th month and total churned accounts in the 7th month .

30. The top 20 ranking of the loss for each product, using CTE and DENSE_RANK.




Commission
New Accounts
Revenue

1. Total Records for each table, identify unique customers & Sales code for sales agent

```
SELECT COUNT(DISTINCT CUST_ACCT) FROM vms.detail_transaksi;
```

Extra options

COUNT(DISTINCT CUST_ACCT)
11080

```
SELECT COUNT(DISTINCT SALES_CODE) FROM vms.agent_sales;
```

Extra options

COUNT(DISTINCT SALES_CODE)
476

2 Total new installations, Group by Department, Which new installation was labelled from the column komisi = “ST”, Ordered by Total new installation from the largest

```
1 SELECT DEPARTMENT,
2        COUNT(DISTINCT CUST_ACCT) AS Total_new_account
3 FROM vms.detail_transaksi
4 WHERE KOMISI ='ST'
5 GROUP BY DEPARTMENT
6 ORDER BY Total_new_account DESC;
```

DEPARTMENT	Total_new_account ▼ 1
SA CIDENG	1066
SA FATMAWATI	870
SA KARAWACI	640
SA SURABAYA	565
SA BEKASI	382
SA CHANNEL	217
SA CIKAMPEK	67
SA SERANG	53
SA BANDUNG	43
SA CIREBON	25

3. Alter column name SALES_CODE into SC in table agent_sales

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0164 seconds.)

```
ALTER TABLE agent_sales CHANGE SALES_CODE SC VARCHAR(5);
```

	SC	DEPARTMENT	POSITION	AGENCY	CITY	PENDIDIKAN
Edit Copy Delete	AB13	SA FATMAWATI	Vendor agent		Jakarta	S1
Edit Copy Delete	AB14	SA FATMAWATI	Vendor agent		Jakarta	S1
Edit Copy Delete	AB16	SA SERANG	Vendor agent		Jakarta	S1
Edit Copy Delete	AB24	SA CHANNEL	Vendor agent		Jakarta	S1
Edit Copy Delete	AB25	SA CIDENG	Vendor agent	AG 2	Jakarta	S1
Edit Copy Delete	AB26	SA CIDENG	Vendor agent	AG 1	Jakarta	S1
Edit Copy Delete	AB5P	SA CIDENG	Vendor agent	AG 1	Jakarta	S1
Edit Copy Delete	AB5V	SA CIDENG	Vendor agent	AG 1	Jakarta	S1
Edit Copy Delete	AB7L	SA CIDENG	Vendor agent	AG 1	Jakarta	S1
Edit Copy Delete	AB7M	SA CIDENG	Vendor agent	AG 1	Jakarta	S1
lit Copy Delete	AB8H	SA FATMAWATI	Vendor agent	AG 1	Jakarta	S1

Console

4. Total: Product, Agency, sales agent, department, city

```sql
SELECT COUNT(DISTINCT ADD_INT_PRODUCT) AS Total_product,
       COUNT(DISTINCT dt.AGENCY) AS Total_agency,
       COUNT(DISTINCT dt.DEPARTMENT) AS Total_department,
       COUNT(DISTINCT SALES_CODE) AS Total_sales,
       COUNT(DISTINCT dt.CITY) AS Total_city
FROM detail_transaksi dt
JOIN agent_sales ag
ON dt.SALES_CODE = ag.SC;
```

Total_product	Total_agency	Total_department	Total_sales	Total_city
46	10	12	463	7

5. Total: New accounts, Revenue, Commission by City, that the new account was labelled by komisi = “ST”

```sql
SELECT CITY,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_revenue,
       FORMAT(SUM(COMMISSION_GROSS),0) AS Total_commission,
       COUNT(CUST_ACCT) AS Total_new_account
FROM detail_transaksi
WHERE KOMISI = 'ST'
GROUP BY CITY;
```

CITY	Total_revenue	Total_commission	Total_new_account
Bandung	13,254,000	14,142,510	43
Cirebon	6,017,000	5,048,485	25
Jakarta	985,118,461	629,356,755	3228
Purwakarta	19,928,333	17,814,382	67
Surabaya	156,198,730	86,808,018	565

6. Group by city for all new account from the largest

```sql
SELECT CITY,
       COUNT(CUST_ACCT) AS Total_new_account
FROM detail_transaksi
WHERE KOMISI='ST'
GROUP BY CITY
ORDER BY Total_new_account DESC;
```

CITY	Total_new_account ▼ 1
Jakarta	3228
Surabaya	565
Purwakarta	67
Bandung	43
Cirebon	25

7. Group by Month of Installation for Account paid Ordered by Revenue From the largest

```sql
SELECT MONTHNAME(INSTALASI_COMPLETED) AS Month_Installation,
       SUM(NILAI_REVENUE_PRODUCT) AS Revenue
FROM detail_transaksi
WHERE YEAR(INSTALASI_COMPLETED) = 2023
      AND KOMISI ='ST'
      AND PAYMENT_STATUS ='PAID'
GROUP BY Month_Installation
ORDER BY Revenue DESC;
```

Month_Installation	Revenue ▼ 1
August	115298855
July	111331731
January	104152705
March	98949113
September	91410024
October	88563647
February	86759413
May	86270415
April	79750681
June	73374064
November	72343666

8. Group by Quarter for Commission Gross From the largest, Trend Commission for new account quarterly

```sql
SELECT
   CASE
     WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
     WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
     WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
     WHEN TERM_PAYMENT IN ('OKT','NOV','DEC') THEN 'Q4'
   END AS Quarter,
   SUM(COMMISSION_GROSS) AS Commission_net
FROM detail_transaksi
WHERE KOMISI = 'ST'
GROUP BY Quarter
ORDER BY Commission_net DESC;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PH

Show all | Number of rows: 25 Filter

Extra options

Quarter	Commission_net 1
Q1	357491142
Q3	354395892
Q2	307154775
Q4	298304289

9. Group by Department for agent sales that has Revenue > Rp.5.000.000 , - Ordered by Revenue from the largest

```sql
SELECT SALES_CODE,
       SUM(NILAI_REVENUE_PRODUCT) AS High_Sales,
       DEPARTMENT,
       CITY,
       AGENCY
FROM detail_transaksi
WHERE KOMISI ='ST'
GROUP BY DEPARTMENT
HAVING High_Sales > 5000000
ORDER BY High_Sales DESC;
```

	SALES_CODE	High_Sales 1	DEPARTMENT	CITY	AGENCY
Edit Copy Delete	KCO5	321852413	SA CIDENG	Jakarta	AG 2
Edit Copy Delete	LOV2	265973560	SA FATMAWATI	Jakarta	AG 1
Edit Copy Delete	LTV4	193266405	SA KARAWACI	Jakarta	AG 1
Edit Copy Delete	KSF1	156198730	SA SURABAYA	Surabaya	AG 1
Edit Copy Delete	LOV8	122407641	SA BEKASI	Jakarta	AG 1
Edit Copy Delete	KYE9	65622917	SA CHANNEL	Jakarta	AG 2
Edit Copy Delete	LWJ6	19928333	SA CIKAMPEK	Purwakarta	AG 1
Edit Copy Delete	KSZ9	15995525	SA SERANG	Jakarta	AG 1
Edit Copy Delete	KBH6	13254000	SA BANDUNG	Bandung	AG 1
Edit Copy Delete	LWR9	6017000	SA CIREBON	Cirebon	AG 1

10. Select Department, Total New Account & Paid revenue, Group By Department that has TOP 5 New Account and ordered By Total new account from the largest

```sql
SELECT DEPARTMENT,
       COUNT(DISTINCT CUST_ACCT) AS Total_new_account,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_revenue
FROM detail_transaksi
WHERE KOMISI ='ST' AND
      PAYMENT_STATUS ='PAID'
GROUP BY DEPARTMENT
ORDER BY Total_new_account DESC
LIMIT 5;
```

DEPARTMENT	Total_new_account ▼ 1	Total_revenue
SA CIDENG	1010	306,052,589
SA FATMAWATI	833	255,823,727
SA KARAWACI	597	180,124,405
SA SURABAYA	518	143,389,397
SA BEKASI	355	114,811,074

11. Average Revenue per sale of each Department from the largest that was calculated by Total Paid Revenue per Total sales agent from the first commission

```sql
SELECT dt.DEPARTMENT,
       COUNT(DISTINCT SALES_CODE) Total_sales_agent,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_revenue,
       ROUND(SUM(NILAI_REVENUE_PRODUCT)/COUNT(DISTINCT SALES_CODE),2)
       AS AVG_REV_per_Sale
FROM detail_transaksi dt
JOIN agent_sales ag ON dt.SALES_CODE = ag.SC
WHERE KOMISI = 'ST'
      AND PAYMENT_STATUS = 'PAID'
GROUP BY dt.DEPARTMENT
ORDER BY AVG_REV_per_Sale DESC;
```

DEPARTMENT	Total_sales_agent	Total_revenue	AVG_REV_per_Sale ▼ 1
SA SURABAYA	10	143,389,397	14338939.70
SA BANDUNG	1	12,741,000	12741000.00
SA CHANNEL	9	63,550,417	7061157.44
SA CIKAMPEK	5	19,173,333	3834666.60
SA FATMAWATI	71	255,823,727	3603151.08
SA BEKASI	37	114,811,074	3103002.00
SA KARAWACI	63	180,124,405	2859117.54
SA SERANG	6	15,462,525	2577087.50
SA CIDENG	129	306,052,589	2372500.69
SA CIREBON	4	5,742,000	1435500.00

12. Select the trend of paid revenue quarterly from the largest. The trend monthly revenue was selected in No.7

```sql
SELECT
   CASE
     WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
     WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
     WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
     ELSE 'Q4'
   END AS Quarter,
   FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_Revenue
FROM detail_transaksi
WHERE KOMISI = 'ST'
      AND PAYMENT_STATUS ='PAID'
GROUP BY Quarter;
```

Quarter	Total_Revenue
Q1	298,293,271
Q2	262,775,209
Q3	302,539,650
Q4	253,262,337

13. Select the trend of New accounts quarterly.

```sql
SELECT
   CASE
     WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
     WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
     WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
     ELSE 'Q4'
   END AS Quarter,
   COUNT(DISTINCT CUST_ACCT) AS Total_new_account
FROM detail_transaksi
WHERE KOMISI = 'ST'
GROUP BY Quarter;
```

Quarter	Total_new_account
Q1	1034
Q2	939
Q3	1058
Q4	897

14. Format-date column: Reported_date, Completed_ install & Churn_Date as (dd/mm/yyyy) in table detail_transaksi was not compatible in phpMYadmin, thus should the format have not been changed into "yyyy-mm-dd", phpMyAdmin could read them as " 0000-00-00", It used Python pandas (Google Collab) to change date format of three columns as follows:

```
[1]  import pandas as pd
     import numpy as np
```

```
[6]  df_x = pd.read_excel('detail_transaksi.xlsx')
```

```
[14] df_x['REPORTED_DATE'] = pd.to_datetime(df_x['REPORTED_DATE'], dayfirst=True).dt.strftime('%Y-%m-%d')
```

```
     df_x['CHURN_DATE'] = pd.to_datetime(df_x['CHURN_DATE'], dayfirst=True, errors='coerce').dt.strftime('%Y-%m-%d')
```

```
[18] df_x['COMPLETED INSTALL'] =pd.to_datetime(df_x['COMPLETED INSTALL'], dayfirst=True).dt.strftime('%Y-%m-%d')
```

```
df_x.to_csv('detail_2.csv', index=False)
```

15. The new table was detail_2

16. Select table detail_2, as it seemed, The Format column of Reported_date ,Completed_install & Churn_date was “yyyy-mm-dd”

	REPORTED_DATE	TERM_PAYMENT	KOMISI	DEPARTMENT	COMPLETED_INSTALL	CUST_ACCT	ADD_INT_PRODUCT
Edit Copy Delete	2023-01-17	JAN	ST	SA SURABAY	2022-12-26	1	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-01	2	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA FATMAWA	2022-12-01	3	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-03	4	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-05	5	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-19	6	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA FATMAWA	2022-12-07	7	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA FATMAWA	2022-12-07	8	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-20	9	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA FATMAWA	2022-12-09	10	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA KARAWAC	2022-12-08	11	PRD 1
Edit Copy Delete	2023-01-17	JAN	ST	SA CIDENG	2022-12-14	12	PRD 1

Console

17. Select the trend of new–monthly - accounts, group & sorted by “Month” (Both PAID and ALL)

```sql
SELECT
  CASE
    WHEN TERM_PAYMENT ='JAN' THEN 1
    WHEN TERM_PAYMENT ='FEB' THEN 2
    WHEN TERM_PAYMENT ='MAR' THEN 3
    WHEN TERM_PAYMENT ='APR' THEN 4
    WHEN TERM_PAYMENT ='MEI' THEN 5
    WHEN TERM_PAYMENT ='JUNE' THEN 6
    WHEN TERM_PAYMENT ='JULY' THEN 7
    WHEN TERM_PAYMENT ='AUG' THEN 8
    WHEN TERM_PAYMENT ='SEP' THEN 9
    WHEN TERM_PAYMENT ='OKT' THEN 10
    WHEN TERM_PAYMENT ='NOV' THEN 11
    ELSE 12
  END AS Month,
  COUNT(DISTINCT CUST_ACCT) AS Total_new_account
  FROM detail_2
  WHERE PAYMENT_STATUS = 'PAID' AND
        KOMISI ='ST'
  GROUP BY Month
  ORDER BY Month;
```

17A. PAID Accounts

Month ▲ 1	Total_new_account
1	370
2	317
3	282
4	319
5	252
6	290
7	258
8	372
9	384
10	331
11	279
12	248

17A. ALL Accounts

Month ▲ 1	Total_new_account
1	397
2	336
3	301
4	360
5	263
6	316
7	268
8	389
9	401
10	348
11	285
12	264

18. Select the trend of Total Commission quarterly from the first, the second and the third one, ordered by quarter

```sql
SELECT
   CASE
     WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
     WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
     WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
     ELSE 'Q4'
   END AS Quarter,
   FORMAT(SUM(COMMISSION_GROSS),0) AS Total_Commission
FROM detail_2
GROUP BY Quarter
ORDER BY Quarter;
```

Quarter ▲ 1	Total_Commission
Q1	634,469,942
Q2	547,871,073
Q3	621,743,008
Q4	574,858,591

19. Select the trend of Total Commission monthly from the first, the second and the third one, ordered by Month

```sql
SELECT
   CASE
     WHEN TERM_PAYMENT IN ('JAN') THEN 1
     WHEN TERM_PAYMENT IN ('FEB') THEN 2
     WHEN TERM_PAYMENT IN ('MAR') THEN 3
     WHEN TERM_PAYMENT IN ('APR') THEN 4
     WHEN TERM_PAYMENT IN ('MEI') THEN 5
     WHEN TERM_PAYMENT IN ('JUNE') THEN 6
     WHEN TERM_PAYMENT IN ('JULY') THEN 7
     WHEN TERM_PAYMENT IN ('AUG') THEN 8
     WHEN TERM_PAYMENT IN ('SEP') THEN 9
     WHEN TERM_PAYMENT IN ('OKT') THEN 10
     WHEN TERM_PAYMENT IN ('NOV') THEN 11
     ELSE 12
   END AS Month,
   FORMAT(SUM(COMMISSION_GROSS),0) AS Total_Commission
FROM detail_2
GROUP BY Month
ORDER BY Month;
```

Month ▲ 1	Total_Commission
1	241,995,445
2	202,740,466
3	189,734,031
4	193,902,827
5	172,249,279
6	181,718,967
7	189,387,703
8	216,776,419
9	215,578,886
10	192,139,683
11	197,329,042
12	185,389,866

Product

- PD1
- PD2
- PD3
- PD7
- PD5

20. Group by product, select the total new accounts with product sold and total revenue, average sales, sorted by total new accounts

```sql
SELECT ADD_INT_PRODUCT AS Product,
       COUNT(CUST_ACCT) AS Total_new_account,
       FORMAT(NILAI_REVENUE_PRODUCT,0) AS Harga_per_product,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_Revenue,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT)/COUNT(CUST_ACCT),0) AS Average_product
FROM detail_2
WHERE KOMISI ='ST' AND
      PAYMENT_STATUS ='PAID'
GROUP BY Product
ORDER BY Total_new_account DESC;
```

Product	Total_new_account ▼ 1	Harga_per_product	Total_Revenue	Average_product
PRD 40	977	275,000	245,053,250	250,822
PRD 2	690	275,000	182,082,234	263,887
PRD 16	666	340,000	212,036,262	318,373
PRD 1	321	238,000	81,452,745	253,747
PRD 39	227	210,000	47,670,000	210,000
PRD 3	185	415,000	73,374,728	396,620
PRD 37	155	210,000	32,550,000	210,000
PRD 42	68	525,000	35,327,375	519,520
PRD 36	66	185,000	12,210,000	185,000
PRD 38	66	185,000	12,210,000	185,000
PRD 27	55	665,000	35,368,169	643,058
PRD 15	54	525,000	26,114,000	483,593
PRD 41	37	695,000	25,765,000	696,351
PRD 11	36	695,000	22,537,338	626,037
PRD 28	23	794,000	22,583,500	981,891
PRD 4	15	245,000	3,627,500	241,833
PRD 7	11	513,700	5,655,416	514,129
PRD 5	11	325,000	3,241,500	294,682
PRD 14	8	395,000	3,090,000	386,250
PRD 29	6	1,798,500	10,972,000	1,828,667
PRD 8	5	782,000	3,858,700	771,740
PRD 6	4	453,800	1,730,000	432,500
PRD 44	4	3,106,500	12,426,000	3,106,500
PRD 43	2	265,000	520,000	260,000
PRD 13	2	238,000	519,500	259,750

Console

21. 10 TOP product stayed ‘ACTIVE’ either for the fourth or the seventh month (from first installation), group by product and ordered by total new accounts from the largest

```sql
SELECT ADD_INT_PRODUCT AS Product,
       COUNT(CUST_ACCT) AS Total_new_account,
       FORMAT(NILAI_REVENUE_PRODUCT,0) AS Harga_per_product,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT),0) AS Total_Revenue,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT)/COUNT(CUST_ACCT),0) AS Average_product
FROM detail_2
WHERE KOMISI ='ND' OR KOMISI ='RD'
GROUP BY Product
ORDER BY Total_new_account DESC
LIMIT 10;
```

Product	Total_new_account ▾ 1	Harga_per_product	Total_Revenue	Average_product
PRD 19	1270	325,500	333,616,256	262,690
PRD 2	1211	275,000	330,914,195	273,257
PRD 4	948	281,500	244,634,831	258,054
PRD 16	785	340,000	257,270,812	327,734
PRD 1	580	238,000	148,610,916	256,226
PRD 40	577	275,000	147,740,000	256,049
PRD 37	324	210,000	68,040,000	210,000
PRD 18	276	238,000	71,795,000	260,127
PRD 3	224	415,000	91,681,702	409,293
PRD 39	148	210,000	31,080,000	210,000

22. Trend of the quarterly most popular product, grouped and ordered by quarter

```sql
WITH CTE1 AS (
    SELECT
      CASE
        WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
        WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
        WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
        ELSE 'Q4'
     END AS Quarter,
     ADD_INT_PRODUCT,
     COUNT(DISTINCT CUST_ACCT) AS Total_new_accounts
    FROM vms.detail_2
    WHERE PAYMENT_STATUS = 'PAID'
          AND kOMISI = 'ST'
    GROUP BY Quarter,ADD_INT_PRODUCT
),
CTE2 AS (
    SELECT
      Quarter,
      ADD_INT_PRODUCT,
      Total_new_accounts,
      DENSE_RANK() OVER (PARTITION BY Quarter ORDER BY Total_new_accounts DESC ) AS rnk
    FROM CTE1
)
SELECT
    Quarter,
    ADD_INT_PRODUCT,
    Total_new_accounts
    FROM CTE2
    WHERE rnk = 1
    GROUP BY Quarter
    ORDER BY Quarter;
```

Quarter	ADD_INT_PRODUCT	Total_new_accounts
Q1	PRD 1	321
Q2	PRD 2	333
Q3	PRD 40	519
Q4	PRD 40	429

23. Trend of the monthly most popular product, grouped by month, ordered by new accounts from the largest

```sql
WITH CTE1 AS (
    SELECT
      TERM_PAYMENT,
      ADD_INT_PRODUCT,
      COUNT(DISTINCT CUST_ACCT) AS Total_new_accounts
    FROM vms.detail_2
    WHERE PAYMENT_STATUS = 'PAID'
          AND kOMISI = 'ST'
    GROUP BY TERM_PAYMENT,ADD_INT_PRODUCT
),
CTE2 AS (
    SELECT
      TERM_PAYMENT,
      ADD_INT_PRODUCT,
      Total_new_accounts,
      DENSE_RANK() OVER (PARTITION BY TERM_PAYMENT ORDER BY Total_new_accounts DESC ) AS rnk
    FROM CTE1
)
SELECT
    TERM_PAYMENT,
    ADD_INT_PRODUCT,
    Total_new_accounts
    FROM CTE2
    WHERE rnk = 1
    GROUP BY TERM_PAYMENT
    ORDER BY Total_new_accounts DESC;
```

TERM_PAYMENT	ADD_INT_PRODUCT	Total_new_accounts
JAN	PRD 1	287
SEP	PRD 40	202
AUG	PRD 40	201
OKT	PRD 40	181
FEB	PRD 2	151
MAR	PRD 2	133
NOV	PRD 40	125
APR	PRD 2	123
DEC	PRD 40	123
JULY	PRD 40	116
JUNE	PRD 2	111
MEI	PRD 2	99

24. Select Profit from the largest (total revenue for 4 months - (total of 1st & 2nd) commission) per product that stayed "active" for 4 months from the first installation, Group by product

```sql
SELECT
    ADD_INT_PRODUCT AS Product,
    FORMAT(NILAI_REVENUE_PRODUCT,0) AS Revenue,
    COUNT(CUST_ACCT) AS Total_Account,
    FORMAT(SUM(COMMISSION_GROSS),0) AS Commission_2,
    FORMAT(SUM(NILAI_REVENUE_PRODUCT *4),0) AS Rev_4_months,
    FORMAT(SUM(COMMISSION_GROSS) + SUM(COMMISSION_GROSS*2),0) AS Total_Com_1_AND_2,
    FORMAT(ROUND(SUM(NILAI_REVENUE_PRODUCT *4)-(SUM(COMMISSION_GROSS) + SUM(COMMISSION_GROSS*2)),2),0) AS Profit
FROM detail_2
WHERE KOMISI = 'ND' AND
      CHURN_DATE = '0000-00-00'
GROUP BY Product
ORDER BY Profit DESC;
```

Product	Revenue	Total_Account	Commission_2	Rev_4_months	Total_Com_1_AND_2	Profit ▼ 1
PRD 37	210,000	191	22,261,050	160,440,000	66,783,150	93,656,850
PRD 31	1,380,750	3	2,298,948	16,569,000	6,896,844	9,672,156
PRD 30	514,500	8	2,217,381	15,981,100	6,652,143	9,328,957
PRD 38	185,000	19	1,950,825	14,060,000	5,852,475	8,207,525

Product	Revenue	Total_Account	Commission_2	Rev_4_months	Total_Com_1_AND_2	Profit 1
PRD 13	325,000	13	1,902,541	13,712,000	5,707,623	8,004,377
PRD 18	281,500	127	18,354,693	132,286,000	55,064,079	77,221,921
PRD 5	290,000	11	1,799,031	12,966,000	5,397,093	7,568,907
PRD 44	3,106,500	1	1,724,107	12,426,000	5,172,321	7,253,679
PRD 12	443,800	7	1,664,737	11,998,100	4,994,211	7,003,889
PRD 27	665,000	37	13,601,198	98,026,668	40,803,594	57,223,074
PRD 39	210,000	104	12,121,200	87,360,000	36,363,600	50,996,400
PRD 15	525,000	43	12,090,120	87,136,000	36,270,360	50,865,640
PRD 19	325,500	80	11,786,682	84,949,000	35,360,046	49,588,954
PRD 2	275,000	670	101,538,911	731,812,056	304,616,733	427,195,323
PRD 14	395,000	5	1,073,925	7,740,000	3,221,775	4,518,225
PRD 6	443,800	4	960,150	6,920,000	2,880,450	4,039,550
PRD 11	695,000	24	9,158,702	66,008,668	27,476,106	38,532,562
PRD 16	340,000	427	77,839,402	561,004,792	233,518,206	327,486,586
PRD 28	975,000	14	7,601,280	54,784,000	22,803,840	31,980,160
PRD 36	185,000	73	7,495,275	54,020,000	22,485,825	31,534,175
PRD 33	849,000	2	891,330	6,424,000	2,673,990	3,750,010
PRD 9	793,000	2	880,230	6,344,000	2,640,690	3,703,310
PRD 34	1,518,750	1	842,906	6,075,000	2,528,718	3,546,282
PRD 17	1,390,750	1	771,866	5,563,000	2,315,598	3,247,402
[illegible]	525,000	24	6,905,586	49,770,000	20,716,758	29,053,242

Console

Continuing no.24

25. Select the best-selling product by department by CTE

```sql
WITH Product_sales AS (
    SELECT
      DEPARTMENT,
      ADD_INT_PRODUCT,
      COUNT(DISTINCT CUST_ACCT) AS Total_account
    FROM detail_2
    WHERE PAYMENT_STATUS ='PAID'
          AND KOMISI = 'ST'
    GROUP BY DEPARTMENT,ADD_INT_PRODUCT
),
Ranked_product AS (
    SELECT
      DEPARTMENT,
      ADD_INT_PRODUCT,
      Total_account,
      RANK() OVER (PARTITION BY DEPARTMENT ORDER BY Total_account DESC) AS Product_rank
    FROM Product_sales
)
SELECT
  DEPARTMENT,
  ADD_INT_PRODUCT,
  Total_account
FROM Ranked_product
WHERE Product_rank = 1
ORDER BY DEPARTMENT;
```

	DEPARTMENT	ADD_INT_PRODUCT	Total_account
Edit Copy Delete	SA BANDUNG	PRD 2	18
Edit Copy Delete	SA BEKASI	PRD 16	88
Edit Copy Delete	SA CHANNEL	PRD 40	58
Edit Copy Delete	SA CIDENG	PRD 40	274
Edit Copy Delete	SA CIKAMPE	PRD 2	33
Edit Copy Delete	SA CIREBON	PRD 37	9
Edit Copy Delete	SA FATMAWA	PRD 40	215
Edit Copy Delete	SA KARAWAC	PRD 40	156
Edit Copy Delete	SA SERANG	PRD 40	12
Edit Copy Delete	SA SURABAY	PRD 40	176

26. Select the best-selling product by city

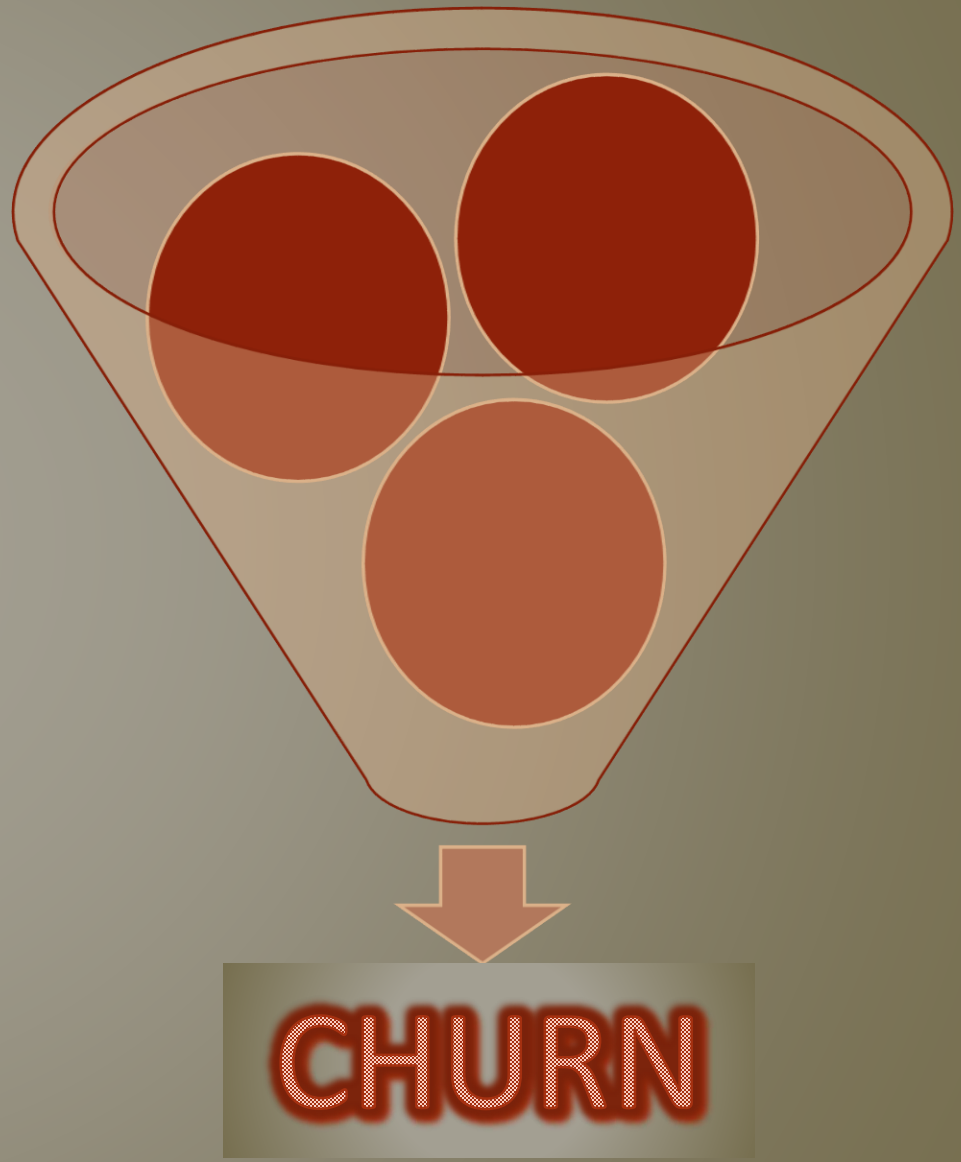
```sql
WITH Productsales AS (
    SELECT
      CITY,
      ADD_INT_PRODUCT,
      COUNT(DISTINCT CUST_ACCT) AS Total_account
    FROM detail_2
    WHERE PAYMENT_STATUS ='PAID'
          AND KOMISI ='ST'
    GROUP BY CITY,ADD_INT_PRODUCT
),
Productrank AS (
    SELECT
      CITY,
      ADD_INT_PRODUCT,
      Total_account,
      RANK() OVER (PARTITION BY CITY ORDER BY Total_account DESC) AS rnk
      FROM Productsales
)
SELECT
  CITY,
  ADD_INT_PRODUCT,
  Total_account
FROM Productrank
WHERE rnk = 1
GROUP BY CITY;
```

	CITY	ADD_INT_PRODUCT	Total_account
Edit Copy Delete	Bandung	PRD 2	18
Edit Copy Delete	Cirebon	PRD 37	9
Edit Copy Delete	Jakarta	PRD 40	798
Edit Copy Delete	Purwakarta	PRD 2	33
Edit Copy Delete	Surabaya	PRD 40	176

27. Select the top 10 best-selling products by sales agent

```sql
WITH Product AS (
    SELECT
        SALES_CODE,
        DEPARTMENT,
        ADD_INT_PRODUCT,
        COUNT(DISTINCT CUST_ACCT) AS Total_new_account
    FROM detail_2
    WHERE PAYMENT_STATUS = 'PAID' AND
          KOMISI = 'ST'
    GROUP BY SALES_CODE,ADD_INT_PRODUCT
),
Rank_product AS (
    SELECT
        SALES_CODE,
        Total_new_account,
        RANK() OVER (PARTITION BY SALES_CODE ORDER BY Total_new_account DESC) AS rnk
    FROM Product
)
SELECT
      a.SALES_CODE,
      b.DEPARTMENT,
      b.ADD_INT_PRODUCT,
      a.Total_new_account
    FROM Rank_product a
    JOIN Product b ON a.SALES_CODE = b.SALES_CODE
    WHERE rnk = 1
    GROUP BY a.SALES_CODE
    ORDER BY a.Total_new_account DESC
    LIMIT 10;
```

SALES_CODE	DEPARTMENT	ADD_INT_PRODUCT	Total_new_account
LBB8	SA SURABAY	PRD 40	56
JTU1	SA BEKASI	PRD 8	53
JXT3	SA SURABAY	PRD 8	52
KDS4	SA SURABAY	PRD 42	43
IBC9	SA FATMAWA	PRD 44	30
KYE9	SA CHANNEL	PRD 42	24
FAY7	SA KARAWAC	PRD 42	24
AD7L	SA CIDENG	PRD 42	22
KVU5	SA CIDENG	PRD 7	22
KCA6	SA CIDENG	PRD 39	21



CHURN

28. Select Total accounts, Total churned accounts, percentage of churn accounts in the second commission that was paid on the 4th month, group by term payment, and ordered by total accounts from the largest

```sql
SELECT
    TERM_PAYMENT,
    COUNT(DISTINCT CUST_ACCT) AS Total_account,
    COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-00 THEN CUST_ACCT END) AS Total_acct_churn,
    (COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-00 THEN CUST_ACCT END)/COUNT(DISTINCT CUST_ACCT)) *100 AS Persentase_account_churn
FROM detail_2
WHERE KOMISI = 'ND'
GROUP BY TERM_PAYMENT
ORDER BY Total_account DESC;
```

TERM_PAYMENT	Total_account ▼ 1	Total_acct_churn	Persentase_account_churn
MAR	377	25	6.6313
NOV	372	17	4.5699
APR	330	11	3.3333
JULY	322	11	3.4161
MEI	287	12	4.1812
SEP	277	13	4.6931
DEC	276	51	18.4783
JUNE	258	8	3.1008
OKT	246	9	3.6585
AUG	231	9	3.8961
JAN	200	7	3.5000
FEB	197	15	7.6142

29. Select Total accounts, Total churned accounts, percentage of churn accounts in the third commission that was paid on the 7th month, group by term payment, and ordered by total accounts from the largest

```sql
SELECT
    TERM_PAYMENT,
    COUNT(DISTINCT CUST_ACCT) AS Total_account,
    COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-00 THEN CUST_ACCT END) AS Total_acct_churn,
    (COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-00 THEN CUST_ACCT END)/COUNT(DISTINCT CUST_ACCT)) *100 AS
Persentase_account_churn
FROM detail_2
WHERE KOMISI = 'RD'
GROUP BY TERM_PAYMENT
ORDER BY Total_account DESC;
```

TERM_PAYMENT	Total_account ▼ 1	Total_acct_churn	Persentase_account_churn
JAN	686	146	21.2828
DEC	384	11	2.8646
JUNE	346	81	23.4104
JULY	330	43	13.0303
OKT	322	61	18.9441
FEB	302	51	16.8874
AUG	287	48	16.7247
SEP	258	34	13.1783
MAR	236	46	19.4915
NOV	231	47	20.3463
APR	200	37	18.5000
MEI	197	50	25.3807

30. Select the top 15 most churned products, revenue, total revenue for 4 months of subscription, total churned accounts and total commission for the 7th month, group by product, and ordered by total churned accounts from the largest

```sql
WITH CTE_1 AS(
   SELECT
       ADD_INT_PRODUCT AS Product,
       FORMAT(NILAI_REVENUE_PRODUCT,0) AS Revenue,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT) * 4,0) AS Total_Rev_4_months,
       COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-00 THEN CUST_ACCT  END) AS Total_acc_churn,
       FORMAT(SUM(NILAI_REVENUE_PRODUCT) * 1.5,0) AS Total_commission

   FROM detail_2
   WHERE KOMISI ='RD'
   GROUP BY Product
)
SELECT
   Product,
   Revenue,
   Total_Rev_4_months,
   Total_acc_churn,
   Total_commission
FROM CTE_1
GROUP BY Product
ORDER BY Total_acc_churn DESC
LIMIT 15;
```

	Product	Revenue	Total_Rev_4_months	Total_acc_churn	Total_commission
Edit Copy Delete	PRD 19	268,750	1,246,716,024	237	467,518,509
Edit Copy Delete	PRD 4	281,500	477,758,500	115	179,159,438
Edit Copy Delete	PRD 2	275,000	541,244,724	97	202,966,772
Edit Copy Delete	PRD 1	238,000	297,221,832	45	111,458,187
Edit Copy Delete	PRD 16	340,000	440,418,456	35	165,156,921
Edit Copy Delete	PRD 37	210,000	99,960,000	35	37,485,000
Edit Copy Delete	PRD 18	281,500	143,590,000	30	53,846,250
Edit Copy Delete	PRD 21	523,000	130,074,000	13	48,777,750
Edit Copy Delete	PRD 36	185,000	39,960,000	11	14,985,000
Edit Copy Delete	PRD 40	245,000	198,515,000	6	74,443,125
Edit Copy Delete	PRD 3	415,000	143,335,604	5	53,750,852
Edit Copy Delete	PRD 22	368,800	17,954,388	4	6,732,896
Edit Copy Delete	PRD 13	245,000	15,012,000	2	5,629,500
Edit Copy Delete	PRD 20	715,300	100,036,400	2	37,513,650
Edit Copy Delete	PRD 26	362,542	11,862,168	2	4,448,313

31. Trend of Total Churned accounts in the 4th, 7th months (2nd,3rd consecutively), and new paid accounts quarterly group and ordered by Quarter

```sql
SELECT

      CASE
        WHEN TERM_PAYMENT IN ('JAN','FEB','MAR') THEN 'Q1'
        WHEN TERM_PAYMENT IN ('APR','MEI','JUNE') THEN 'Q2'
        WHEN TERM_PAYMENT IN ('JULY','AUG','SEP') THEN 'Q3'
        ELSE 'Q4'
      END AS Quarter,
      COUNT(DISTINCT CASE WHEN KOMISI = 'ND' AND CHURN_DATE <>0000-00-00 THEN CUST_ACCT END)
AS Total_account_2nd,
      COUNT(DISTINCT CASE WHEN KOMISI = 'RD' AND CHURN_DATE <> 0000-00-00 THEN CUST_ACCT END)
AS Total_account_3rd,
      COUNT(DISTINCT CASE WHEN KOMISI = 'ST' THEN CUST_ACCT END) AS Total_account_1st
   FROM detail_2
   WHERE PAYMENT_STATUS ='PAID'
   GROUP BY Quarter;
```

Quarter	Total_churned_account_2nd	Total_churned_account_3rd	Total_account_1st
Q1	47	243	969
Q2	31	168	861
Q3	33	125	1014
Q4	77	119	858

Show all | Number of rows: 25 Filter rows: Search this table

32. Group by Department, select the total new paid accounts, total churned accounts in the 4th month (2nd), and total churned accounts in the 7th month (3rd)

```sql
SELECT
  DEPARTMENT,
  COUNT(DISTINCT CASE WHEN KOMISI ='ST' THEN CUST_ACCT END) AS Total_new_acc,
  COUNT(DISTINCT CASE WHEN KOMISI = 'ND' AND CHURN_DATE <>0000-00-00 THEN CUST_ACCT END) AS Total_churned_acc_2nd,
  COUNT(DISTINCT CASE WHEN KOMISI = 'RD' AND CHURN_DATE <>0000-00-00 THEN CUST_ACCT END) AS Total_churned_acc_3rd
FROM detail_2
WHERE PAYMENT_STATUS ='PAID'
GROUP BY DEPARTMENT
ORDER BY DEPARTMENT;
```

DEPARTMENT ▲ 1	Total_new_acc	Total_churned_acc_2nd	Total_churned_acc_3rd
SA BALI	0	0	1
SA BANDUNG	41	2	34
SA BEKASI	355	15	59
SA CHANNEL	209	8	35
SA CIDENG	1010	46	174
SA CIKAMPE	64	4	5
SA CIREBON	24	4	22
SA FATMAWA	833	33	124
SA KARAWAC	597	28	88
SA SEMARAN	0	0	1
SA SERANG	51	2	9
SA SURABAY	518	46	103

33a. Write a query to analyse and order by the top 20 ranking of the loss for each product, using CTE and DENSE_RANK to select columns: Revenue per product, Revenue for 6 months, all accounts in either the 4th or 7th month. Total churned accounts, the remaining from inactive months, the loss, and the ranking.

```
1  WITH CTE1 AS(
2  SELECT
3         ADD_INT_PRODUCT AS Product,
4         NILAI_REVENUE_PRODUCT AS Revenue,
5         SUM(NILAI_REVENUE_PRODUCT) * 6 AS Revenue_6_months,
6         COUNT(DISTINCT CUST_ACCT) AS All_accounts,
7         COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-0 THEN CUST_ACCT END) AS Total_account_churn,
8         SUM(-ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0)) AS Total_active_months,
9         (COUNT(DISTINCT CASE WHEN CHURN_DATE <> 0000-00-0 THEN CUST_ACCT END)*6) - SUM(-
   ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0)) AS Inactive_months
10 FROM detail_2
11 WHERE CHURN_DATE <> 0000-00-00 OR KOMISI <> 'ST'
12 GROUP BY Product
13 ),
14 CTE2 AS (
```

33b. Continuing query

```
),
CTE2 AS (
    SELECT
        Product,
        Inactive_months,
        Inactive_months * Revenue AS The_loss
    FROM CTE1
    GROUP BY Product, Inactive_months
),
CTE3 AS (
    SELECT
        Product,
        The_loss,
        DENSE_RANK() OVER  (ORDER BY The_loss DESC) AS Ranking
    FROM CTE2
)
SELECT
  b.Product,
  a.Revenue,
  a.Revenue_6_months,
  a.All_accounts,
  a.Total_account_churn,
  a.Total_active_months,
  a.Inactive_months,
  FORMAT(b.The_loss,0) AS The_loss,
  b.ranking

FROM CTE1 a JOIN CTE3 b ON a.Product = b.Product
WHERE Ranking
ORDER BY Ranking ASC
LIMIT 20;
```

Product	Revenue	Revenue_6_months	All_accounts	Total_account_churn	Total_active_months	Inactive_months	The_loss	ranking
PRD 19	325500	2001697536	1270	239	1180	254	82,677,000	1
PRD 2	275000	1985485170	1211	143	606	252	69,300,000	2
PRD 4	238000	1467808986	948	148	624	264	62,832,000	3
PRD 16	340000	1543624872	785	56	229	107	36,380,000	4
PRD 1	238000	891665496	580	56	231	105	24,990,000	5
PRD 40	275000	886440000	577	28	87	81	22,275,000	6
PRD 18	238000	430770000	276	41	155	91	21,658,000	7
PRD 21	513700	238034400	75	13	57	21	10,787,700	8
PRD 37	210000	408240000	324	49	247	47	9,870,000	9
PRD 28	975000	169515000	29	2	2	10	9,750,000	10
PRD 29	1798500	108996000	10	1	1	5	8,992,500	11
PRD 15	525000	253554000	82	8	31	17	8,925,000	12
PRD 3	415000	550090212	224	10	40	20	8,300,000	13
PRD 11	695000	189686004	46	2	4	8	5,560,000	14
PRD 8	772000	135891492	30	3	11	7	5,404,000	15
PRD 22	368800	26931582	14	4	15	9	3,319,200	16
PRD 39	210000	186480000	148	6	22	14	2,940,000	17
PRD 36	185000	144300000	130	14	69	15	2,775,000	18
PRD 7	691200	119504148	37	2	8	4	2,764,800	19
PRD 42	525000	123764250	40	2	7	5	2,625,000	20

34. To compare 20 TOP churned accounts month by month, group by product, and order by total churned accounts

```sql
WITH C1 AS (
SELECT
   ADD_INT_PRODUCT AS Product,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 1 THEN CUST_ACCT END) AS inactive_1st,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 2 THEN CUST_ACCT END) AS inactive_2nd,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 3 THEN CUST_ACCT END) AS inactive_3rd,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 4 THEN CUST_ACCT END) AS inactive_4th,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 5 THEN CUST_ACCT END) AS inactive_5th,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 6 THEN CUST_ACCT END) AS inactive_6th,
   COUNT(DISTINCT CASE WHEN -ROUND(DATEDIFF(COMPLETED_INSTALL,CHURN_DATE)/30.44,0) = 7 THEN CUST_ACCT END) AS inactive_7th

FROM vms.detail_2
WHERE KOMISI IN ('ND','RD') OR
            CHURN_DATE <> 0000-00-00
GROUP BY Product
),
C2 AS (
 SELECT
      Product,
  (inactive_1st+inactive_2nd+inactive_3rd+inactive_4th+inactive_5th+inactive_6th+inactive_7th) AS Total
    FROM C1
)
SELECT
  a.Product,
  a.inactive_1st,
  a.inactive_2nd,
  a.inactive_3rd,
  a.inactive_4th,
  a.inactive_5th,
  a.inactive_6th,
  a.inactive_7th,
  b.Total
FROM C2 b JOIN C1 a ON a.Product=b.Product
GROUP BY Product
Order BY Total DESC
LIMIT 20;
```

Returning query No.34

Product	inactive_1st	inactive_2nd	inactive_3rd	inactive_4th	inactive_5th	inactive_6th	inactive_7th	Total
PRD 19	14	6	13	35	78	66	27	239
PRD 4	8	20	30	18	29	32	11	148
PRD 2	14	16	18	24	28	31	12	143
PRD 16	3	11	8	9	11	9	5	56
PRD 1	4	8	4	13	17	8	2	56
PRD 37	2	6	0	6	7	22	6	49
PRD 18	4	10	8	1	6	11	1	41
PRD 40	4	6	11	1	2	4	0	28
PRD 36	0	0	2	3	3	6	0	14
PRD 21	0	2	1	4	3	2	1	13
PRD 3	0	0	6	1	1	1	1	10
PRD 15	0	0	6	0	0	1	1	8
PRD 39	0	2	2	0	0	2	0	6
PRD 22	0	1	0	2	1	0	0	4
PRD 13	0	0	2	0	0	1	0	3
PRD 8	0	0	2	0	1	0	0	3
PRD 20	0	0	0	0	1	1	0	2
PRD 7	0	0	0	2	0	0	0	2
PRD 5	0	0	0	0	0	0	2	2
PRD 23	0	0	0	0	0	1	1	2

INSIGHT REPORT

1. The months contributed the largest paid accounts: in September, August, and January, and August, July, and January for revenue, quarterly best revenue in Q3, around 302.339.650, with 1058 accounts.

2. The standout products that have role to reach those numbers were Product 40 with 384 & 372 and Product with 370 new paid accounts in September, August and January respectively, Product 40 (revenue 275k) contributed 977 total paid accounts, following Product 2(revenue 275k) and Product 16 (revenue 340k) with 690 and 666 paid accounts in the entire year.

3. Whereas, the best profitability of a product, calculated from the revenue per product that was shown by Product 44 (3.106.500), Product 29 (1.828.667), and Product 28 (981.891).

4. The quarterly best-selling products were Q1 (Product 1), Q2 (Product 2), Q3 & Q4 (Product 40) to new paid accounts.

5. The largest percentages of churned accounts to active accounts in the 4th month of subscription were shown in December, February, and March, roughly 18%, 8%, and 7%, respectively, in the 7th month the percentages increased into 25%, 23% and 21% in Mei, June, January, the TOP 2 best-churned products to disconnected accounts were Product 19 (revenue 268.750) and Product 4 (revenue 281.500) with 239 and 148 accounts.

6. Hence, the total quarterly churned accounts for two stints were Q1 (30%), Q2 and Q4 (23%), and Q3 (16%) in the entire year.

7. Month-over-month development of total churned accounts by product, averaged disconnection rates rose sharply from the 1st month to the 7th month of subscription.

CONCLUSION AND CLOSURE

Profit is not only the accumulated product sales, product development with a higher rate of profitability needs to be prioritized and maximized. Furthermore, to re-evaluate the high churn rate by examining customer segmentation, the service, agents' selling skill have to be nurtured and strengthened, as well as other factors affecting the company's and product's brand reputation. The focus should not be only on fixes and improvements, but also on addressing the underlying causes..

These insights provide the solid details for strategic planning and operational improvements in the upcoming quarters.

Tools: Web-based MySQL database management tool, PYTHON Pandas Google Colab, and PowerPoint.

TERIMA KASIH

By Ainun Jariyah