**Продажи магазина напитков**

-- BEGINNER QUESTIONS --

**1) Наиболее покупаемые продукты (топ 10)**

SELECT

Product,

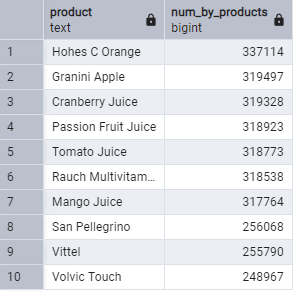
COUNT(\*) AS Num\_by\_products

FROM beverage

GROUP BY Product

ORDER BY num\_by\_products DESC

LIMIT 10;



**2) Количество покупок по категориям**

SELECT

Category,

COUNT(\*) AS Num\_by\_categories

FROM beverage

GROUP BY Category

ORDER BY num\_by\_categories DESC;



**3) Наибольшее количество заказов по регионам (топ 5)**

SELECT

Region,

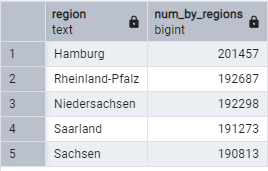
COUNT(DISTINCT Order\_ID) AS Num\_by\_regions

FROM beverage

GROUP BY Region

ORDER BY num\_by\_regions DESC

LIMIT 5;



**4) Количество заказов по типам клиентов**

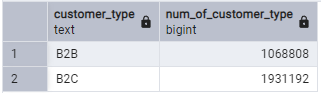
SELECT

Customer\_Type,

COUNT(DISTINCT Order\_ID) AS Num\_of\_customer\_type

FROM beverage

GROUP BY Customer\_Type;



-- INTERMEDIATE QUESTIONS --

**1) Затраты и количество купленных товаров по регионам**

SELECT

Region,

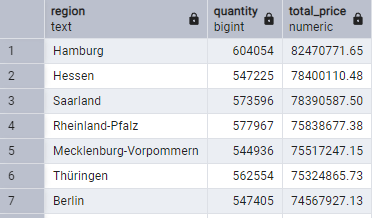
COUNT(Quantity) AS Quantity,

ROUND(SUM(Total\_Price::numeric), 2) AS Total\_Price

FROM beverage

GROUP BY Region

ORDER BY Total\_Price DESC;



**2) Затраты и количество купленных товаров по категориям**

SELECT

Category,

COUNT(Quantity) AS Quantity,

ROUND(SUM(Total\_Price::numeric), 2) AS Total\_Price

FROM beverage

GROUP BY Category

ORDER BY Total\_Price DESC;



**3) Продажи по дате (по дням)**

SELECT

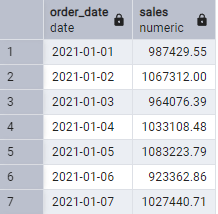
Order\_Date,

ROUND(SUM(Total\_Price::numeric), 2) AS Sales

FROM beverage

GROUP BY Order\_Date

ORDER BY Order\_Date;



**4) Продажи по дате (по месяцам)**

SELECT DATE\_PART('year', Order\_Date) AS date\_year,

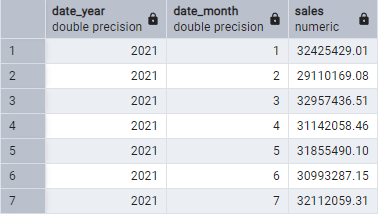
DATE\_PART('month', Order\_Date) AS date\_month,

ROUND(SUM(Total\_Price::numeric), 2) AS Sales

FROM beverage

GROUP BY date\_month, date\_year

ORDER BY date\_year, date\_month;



**5) Прибыль по дате (по годам)**

SELECT

DATE\_PART('year', Order\_Date) AS Sales\_by\_Year,

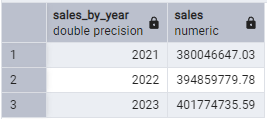
ROUND(SUM(Total\_Price::numeric), 2) AS Sales

FROM beverage

GROUP BY Sales\_by\_Year

ORDER BY Sales\_by\_Year

LIMIT 10;



-- МЕТРИЧЕСКИЕ ПОКАЗАТЕЛИ --

**1) DAU: число покупателей в день (за последние полгода)**

WITH d AS(

SELECT

Order\_Date,

COUNT(DISTINCT Customer\_ID) AS cust\_day

FROM beverage

WHERE Order\_Date >= '2023-07-01'

GROUP BY Order\_Date

)

SELECT

ROUND(AVG(cust\_day)::numeric, 2) AS DAU

FROM d;



**2) WAU: число покупателей в неделю (за последние полгода)**

WITH w AS(

SELECT

DATE\_PART('year', Order\_Date) AS year\_num,

DATE\_PART('week', Order\_Date) AS week\_num,

COUNT(DISTINCT Customer\_ID) AS cust\_week

FROM beverage

WHERE Order\_Date >= '2023-07-01'

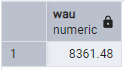
GROUP BY year\_num, week\_num

)

SELECT

ROUND(AVG(cust\_week)::numeric, 2) AS WAU

FROM w;



**3) MAU: число покупателей в месяц (за последние полгода)**

WITH m AS(

SELECT

DATE\_PART('year', Order\_Date) AS year\_num,

DATE\_PART('month', Order\_Date) AS month\_num,

COUNT(DISTINCT Customer\_ID) AS cust\_month

FROM beverage

WHERE Order\_Date >= '2023-07-01'

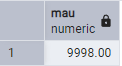
GROUP BY year\_num, month\_num

)

SELECT

ROUND(AVG(cust\_month)::numeric, 2) AS MAU

FROM m;



**4) Sticky\_factor: степень лояльности и частота взаимодействия клиентов (за последние полгода)**

WITH d AS(

SELECT

Order\_Date,

COUNT(DISTINCT Customer\_ID) AS cust\_day

FROM beverage

WHERE Order\_Date >= '2023-07-01'

GROUP BY Order\_Date

),

m AS(

SELECT

DATE\_PART('year', Order\_Date) AS year\_num,

DATE\_PART('month', Order\_Date) AS month\_num,

COUNT(DISTINCT Customer\_ID) AS cust\_month

FROM beverage

WHERE Order\_Date >= '2023-07-01'

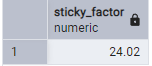
GROUP BY year\_num, month\_num

)

SELECT

ROUND(AVG(cust\_day) / AVG(cust\_month) \* 100, 2) AS sticky\_factor

FROM d, m;



**5) Lifetime**

WITH f as( -- День первой покупки

SELECT DISTINCT Customer\_ID,

Order\_Date,

FIRST\_VALUE(Order\_Date) OVER (PARTITION BY Customer\_ID ORDER BY Order\_Date) AS First\_Order

FROM beverage

WHERE Order\_Date >= '2023-07-01'

ORDER BY Order\_Date

),

df as ( -- Разница между датой текущей покупки и датой первой покупки

SELECT

Order\_Date - First\_Order AS diff,

COUNT(DISTINCT Customer\_ID) AS count\_cust

FROM f

GROUP BY diff

),

ret as( -- Доля оставшихся клиентов

SELECT

diff,

ROUND(count\_cust \* 1.0 / FIRST\_VALUE(count\_cust) OVER (ORDER BY diff), 4) AS retention

FROM df

)

SELECT

SUM(retention) AS lifetime

FROM ret;

