

1. Найти регионы с покупкой более чем одним типом товаров

```
SELECT
    region,
    COUNT(DISTINCT itemname) AS unique_itemname_count
FROM
    sales
GROUP BY
    region
HAVING
    COUNT(DISTINCT itemname) > 1
LIMIT
    100;
```

Результат:

region	unique_itemname_count
Москва Восток	2
Москва Запад	2
Москва Север	2
Москва Центр	2

2. Найти среднюю стоимость 2х наиболее часто покупаемых типов товаров

```
WITH t1 AS (
    SELECT
        itemname,
        COUNT(qty) AS count_qty,
        ROUND(AVG(netsales)::DECIMAL, 2) AS avg_netsales
    FROM sales
    GROUP BY itemname
    ORDER BY count_qty DESC
    /*сортируем по убыванию,
    то есть самые 'покупаемые' сверху */
    LIMIT 2 -- выбираем только 2 самых верхних
)
SELECT itemname, avg_netsales
FROM t1;
```

itemname	avg_netsales
Coffee Columbia	3773.82
Coffee Japan	1559.87

### 3. Рассчитать среднее время (кол-во дней) между покупками в каждом регионе

```
WITH t1 AS (  
    SELECT  
        region,  
        actiondate,  
        LAG(actiondate, 1) OVER (PARTITION BY region ORDER BY actiondate) AS lag_actiondate  
    FROM  
        sales  
) -- создаем колонку со сдвигом в 1 строку, для дальнейшего сравнения даты заказов  
SELECT  
    region,  
    DATE_PART('day', AVG(AGE(actiondate, lag_actiondate))) AS days_between_purchases  
    /*тут считаем дни с помощью age(). считаем среднее,  
    выводим только целые дни с помощью date_part() */  
FROM  
    t1  
WHERE  
    lag_actiondate IS NOT NULL  
GROUP BY  
    region  
ORDER BY  
    days_between_purchases DESC;
```

region	days_between_purchases
Москва Центр	14
Москва Восток	6
Москва Запад	2
Москва Север	1
Москва Юг	1

4. Посчитать процент **GMV** (оборот) кофе **japan** от всех покупок в каждом регионе за каждый месяц

```
with t1 as (  
select  
    region,  
    itemname,  
    netsales,  
    date_trunc('month',actiondate)::date as month_actiondate,  
    sum(netsales) over(partition by  
        region,  
        date_trunc('month', actiondate)  
        order by date_trunc('month',actiondate)  
    ) as sum_by_month --тут считаем сумму покупок, партируем ее на регионы и на месяцы  
from sales  
order by region)  
select  
    region,  
    itemname,  
    month_actiondate,  
    round(sum(netsales / sum_by_month)::decimal,2) * 100 as gmv  
from t1  
where itemname = 'Coffee Japan' --если убрать фильтр можем увидеть как все виды --товаров  
--складываются по месяцам и регионам в 100%  
group by region,itemname,month_actiondate  
order by month_actiondate
```

region	itemname	month_actiondate	gmv
Москва Восток	Coffee Japan	2022-06-01	15
Москва Запад	Coffee Japan	2022-06-01	17
Москва Запад	Coffee Japan	2022-07-01	11
Москва Север	Coffee Japan	2022-07-01	36

5. Отсортировать по дате покупок и для первых 5 строчек посчитать скользящее среднее **NetSales** в размером окна 3 - т.е. средней из текущей, предыдущей и следующей строчек

```
SELECT
    itemname,
    actiondate,
    netsales,
    ROUND(
        AVG(netsales) OVER (
            ORDER BY actiondate
            ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING)::DECIMAL, 2) AS moving_avg
    -- берем среднее из 3х значений
FROM sales
LIMIT 5;
```

itemname	actiondate	netsales	moving_avg
Coffee Japan	2022-06-02	1290.0	6708.33
Coffee Columbia	2022-06-04	12126.67	5482.78
Coffee Columbia	2022-06-05	3031.67	6063.33
Coffee Columbia	2022-06-08	3031.67	2875.64
Coffee Japan	2022-06-20	2563.6	2292.36

6. За каждый день вывести канал привлечения с максимальным **GMV**, и собственно сам **GMV** этого канала

```
WITH t1 AS (  
    SELECT  
        actiondate,  
        medium,  
        ROUND(SUM(netsales * qty)::DECIMAL, 2) AS GMV,  
        RANK() OVER (PARTITION BY actiondate ORDER BY SUM(netsales * qty) DESC) AS rank_medium  
        -- присваиваем рейтинг каналу привлечения, где 1 это самая большая выручка.  
    FROM sales  
    GROUP BY  
        actiondate,  
        medium  
)  
SELECT  
    actiondate,  
    medium,  
    GMV  
FROM t1  
WHERE rank_medium = 1  
ORDER BY GMV DESC  
LIMIT 100;
```

	actiondate	medium	gmv
1	2022-06-30	Advertising	56875.00
2	2022-07-23	Advertising	49979.17
3	2022-06-04	Advertising	48506.67
4	2022-07-13	Advertising	34653.33
5	2022-07-18	Advertising	31986.67
6	2022-07-02	Advertising	20797.50
7	2022-07-20	Referral	17992.50
8	2022-07-05	Advertising	11554.17
9	2022-06-26	Advertising	10916.67
10	2022-06-28	Referral	9626.67
11	2022-08-28	Referral	9360.00
12	2022-07-04	Advertising	9243.33
13	2022-07-26	Organic	7996.67
14	2022-07-26	Advertising	7996.67
15	2022-07-17	Referral	7996.67
16	2022-07-22	Referral	7346.47
17	2022-07-15	Advertising	5265.81
18	2022-07-31	Organic	5245.00
19	2022-06-27	Advertising	5160.00
20	2022-06-20	Advertising	5127.20
21	2022-07-09	Advertising	4526.63
22	2022-07-11	Advertising	4441.67
23	2022-07-16	Advertising	3998.33
24	2022-07-12	Advertising	3433.31
25	2022-08-29	Referral	3120.00

26	2022-06-08	Advertising	3031.67
27	2022-06-05	Advertising	3031.67
28	2022-06-29	Referral	2406.67
29	2022-07-06	Advertising	2310.83
30	2022-07-10	Advertising	2275.83
31	2022-07-07	Advertising	2250.79
32	2022-07-08	Advertising	2250.79
33	2022-07-14	Advertising	2165.83
34	2022-07-21	Organic	1999.17
35	2022-07-19	Advertising	1999.17
36	2022-07-27	Organic	1999.17
37	2022-07-24	Referral	1836.62
38	2022-06-02	Organic	1290.00
39	2022-06-21	Advertising	1281.80

7. Найти наименее маржинальный канал привлечения (для которого минимально отношение прибыли к выручке)

```
select
    medium,
    round((sum(profit) / sum(netsales) * 100)::decimal,2) as profit_margin_percentage
    -- считаем всю сумму профита и делим на всю сумму выручки
from sales
group by medium --разделяем на каналы и выбираем наименьший результат
order by 2
limit 1
```

medium	profit_margin_percentage
Organic	2.53159