Q.1: Assume you're given a table containing information on Facebook user actions. Write a query to obtain the number of monthly active users (MAUs) in July 2022, including the month in numerical format "1, 2, 3".
Hint:

- An active user is defined as a user who has performed actions such as 'sign-in', 'like', or 'comment' in both the current month and the previous month.

`user_actions` Table:

Column Name	Type
user_id	integer
event_id	integer
event_type	string ("sign-in, "like", "comment")
event_date	datetime

`user_actions`Example Input:

user_id	event_id	event_type	event_date
445	7765	sign-in	05/31/2022 12:00:00
742	6458	sign-in	06/03/2022 12:00:00
445	3634	like	06/05/2022 12:00:00
742	1374	comment	06/05/2022 12:00:00
648	3124	like	06/18/2022 12:00:00

Example Output for June 2022:

month	monthly_active_users
6	1

Example

In June 2022, there was only one monthly active user (MAU) with the `user_id` 445.

Please note that the output provided is for June 2022 as the `user_actions` table only contains event dates for that month. You should adapt the solution accordingly for July 2022.

Q.2: Assume you are given the table below containing information on user transactions for particular products. Write a query to obtain the year-on-year growth rate for the total spend of each product for each year.
Output the year (in ascending order) partitioned by product id, current year's spend, previous year's spend and year-on-year growth rate (percentage rounded to 2 decimal places).

`user_transactions` Table:

Column Name	Type
transaction_id	integer
product_id	integer
spend	decimal
transaction_date	datetime

`user_transactions` Example Input:

transaction_id	product_id	spend	transaction_date
1341	123424	1500.60	12/31/2019 12:00:00
1423	123424	1000.20	12/31/2020 12:00:00
1623	123424	1246.44	12/31/2021 12:00:00
1322	123424	2145.32	12/31/2022 12:00:00

Example Output:

year	product_id	curr_year_spend	prev_year_spend	yoy_rate
2019	123424	1500.60		
2020	123424	1000.20	1500.60	-33.35
2021	123424	1246.44	1000.20	24.62
2022	123424	2145.32	1246.44	72.12

The third row in the example output shows that the spend for product 123424 grew 24.62% from 1000.20 in 2020 to 1246.44 in 2021.

Q.3: Amazon wants to maximise the number of items it can stock in a 500,000 square feet warehouse. It wants to stock as many prime items as possible, and afterwards use the remaining square footage to stock the most number of non-prime items.
Write a query to find the number of prime and non-prime items that can be stored in the 500,000 square feet warehouse. Output the item type with `prime_eligible` followed by `not_prime` and the maximum number of items that can be stocked.

Assumptions:

- Prime and non-prime items have to be stored in equal amounts, regardless of their size or square footage. This implies that prime items will be stored separately from non-prime items in their respective containers, but within each container, all items must be in the same amount.
- Non-prime items must always be available in stock to meet customer demand, so the non-prime item count should never be zero.
- Item count should be whole numbers (integers).

`inventory` table:

Column Name	Type
item_id	integer
item_type	string
item_category	string
square_footage	decimal

`inventory` Example Input:

item_id	item_type	item_category	square_footage
1374	prime_eligible	mini refrigerator	68.00
4245	not_prime	standing lamp	26.40
2452	prime_eligible	television	85.00
3255	not_prime	side table	22.60
1672	prime_eligible	laptop	8.50

Example Output:

item_type	item_count
prime_eligible	9285

not_prime	6

Q.4: Google's marketing team is making a Superbowl commercial and needs a simple statistic to put on their TV ad: the median number of searches a person made last year.
However, at Google scale, querying the 2 trillion searches is too costly. Luckily, you have access to the summary table which tells you the number of searches made last year and how many Google users fall into that bucket.

Write a query to report the median of searches made by a user. Round the median to one decimal point.

`search_frequency` Table:

Column Name	Type
searches	integer
num_users	integer

`search_frequency` Example Input:

searches	num_users
1	2
2	2
3	3
4	1

Example Output:

median
2.5

By expanding the search_frequency table, we get [1, 1, 2, 2, 3, 3, 3, 4] which has a median of 2.5 searches per user.

Q.5: Write a query to update the Facebook advertiser's status using the `daily_pay` table. `Advertiser` is a two-column table containing the user id and their payment status based on the last payment and `daily_pay` table has current information about their payment. Only advertisers who paid will show up in this table.
Output the user id and current payment status sorted by the user id.

`advertiser` Table:

Column Name	Type
user_id	string
status	string

`advertiser` Example Input:

user_id	status
bing	NEW
yahoo	NEW
alibaba	EXISTING

`daily_pay` Table:

Column Name	Type
user_id	string
paid	decimal

`daily_pay` Example Input:

user_id	paid
yahoo	45.00
alibaba	100.00
target	13.00

Definition of advertiser status:

- **New**: newly registered users who made their first payment.
- **Existing**: users who paid previously and recently made a current payment.
- **Churn**: users who paid previously, but have yet to make any recent payment.
- **Resurrect**: users who did not pay recently but may have made a previous payment and have made payment again recently.

Example Output:

user_id	new_status
bing	CHURN
yahoo	EXISTING
alibaba	EXISTING

Bing's updated status is CHURN because no payment was made in the `daily_pay` table whereas Yahoo which made a payment is updated as EXISTING.

Q.6: You’re a consultant for a major pizza chain that will be running a promotion where all 3-topping pizzas will be sold for a fixed price, and are trying to understand the costs involved. Given a list of pizza toppings, consider all the possible 3-topping pizzas, and print out the total cost of those 3 toppings. Sort the results with the highest total cost on the top followed by pizza toppings in ascending order.

Break ties by listing the ingredients in alphabetical order, starting from the first ingredient, followed by the second and third.

P.S. Be careful with the spacing (or lack of) between each ingredient. Refer to our Example Output.

Notes:

- Do not display pizzas where a topping is repeated. For example, ‘Pepperoni,Pepperoni,Onion Pizza’.
- Ingredients must be listed in alphabetical order. For example, 'Chicken,Onions,Sausage'. 'Onion,Sausage,Chicken' is not acceptable.

`pizza_toppings` Table:

Column Name	Type
topping_name	varchar(255)
ingredient_cost	decimal(10,2)

`pizza_toppings` Example Input:

topping_name	ingredient_cost
Pepperoni	0.50
Sausage	0.70

Chicken	0.55
Extra Cheese	0.40

Example Output:

pizza	total_cost
Chicken,Pepperoni,Sausage	1.75
Chicken,Extra Cheese,Sausage	1.65
Extra Cheese,Pepperoni,Sausage	1.60
Chicken,Extra Cheese,Pepperoni	1.45

Q.7: UnitedHealth Group has a program called Advocate4Me, which allows members to call an advocate and receive support for their health care needs – whether that's behavioural, clinical, well-being, health care financing, benefits, claims or pharmacy help.
Write a query to get the patients who made a call within 7 days of their previous call. If a patient called more than twice in a span of 7 days, count them as once.

`callers` Table:

Column Name	Type
policy_holder_id	integer
case_id	varchar
call_category	varchar
call_received	timestamp
call_duration_secs	integer
original_order	integer

`callers` Example Input:

policy_holder_id	case_id	call_category	call_received	call_duration_secs	original_order
50837000	dc63-acae-4f39-bb04	claims	3/9/2022 2:51	205	130
50837000	41be-bebe-4bd0-a1ba	IT_support	3/12/2022 5:37	254	129
50837000	bab1-3ec5-4867-90ae	benefits	5/13/2022 18:19	228	339
50936674	12c8-b35c-48a3-b38d	claims	5/31/2022 7:27	240	31
50886837	d0b4-8ea7-4b8c-aa8b	IT_support	3/11/2022 3:38	276	16
50886837	a741-c279-41c0-90ba		3/19/2022 10:52	131	325

Example Output:

patient_count
1

Q.8: UnitedHealth Group has a program called Advocate4Me, which allows members to call an advocate and receive support for their health care needs – whether that's behavioural, clinical, well-being, health care financing, benefits, claims or pharmacy help.
A long-call is categorised as any call that lasts more than 5 minutes (300 seconds). What's the month-over-month growth of long-calls?

Output the year, month (both in numerical and chronological order) and growth percentage rounded to 1 decimal place.

`callers` Table:

Column Name	Type
policy_holder_id	integer
case_id	varchar
call_category	varchar
call_received	timestamp
call_duration_secs	integer
original_order	integer

`callers` Example Input:

policy_holder_id	case_id	call_category	call_received	call_duration_secs	original_order
50986511	b274-c8f0-4d5c-8704		2022-01-28T09:46:00	252	456
54026568	405a-b9be-45c2-b311	n/a	2022-01-29T16:19:00	397	217
54026568	c4cc-fd40-4780-8a53	benefits	2022-01-30T08:18:00	320	134
54026568	81e8-6abf-425b-add2	n/a	2022-02-20T17:26:00	1324	83

54475101	5919-b9c2-49a5-8091		2022-02-24T18:07:00	206	498
54624612	a17f-a415-4727-9a3f	benefits	2022-02-27T10:56:00	435	19
53777383	dfa9-e5a7-4a9b-a756	benefits	2022-03-19T00:10:00	318	69
52880317	cf00-56c4-4e76-963a	claims	2022-03-21T01:12:00	340	254
52680969	0c3c-7b87-489a-9857		2022-03-21T14:00:00	310	213
54574775	ca73-bf99-46b2-a79b	billing	2022-04-18T14:09:00	181	312
51435044	6546-61b4-4a05-9a5e		2022-04-18T21:58:00	354	439
52780643	e35a-a7c2-4718-a65d	n/a	2022-05-06T14:31:00	318	186
54026568	61ac-eee7-42fa-a674		2022-05-07T01:27:00	404	341
54674449	3d9d-e6e2-49d5-a1a0	billing	2022-05-09T11:00:00	107	450

54026568	c516-0063-4b8f-aa74		2022-05-13T01:06:00	404	270

Example Output:

yr	mth	growth_pct
2022	1	0
2022	2	0
2022	3	50.0
2022	4	-66.7
2022	5	200.0

Q.9: Sometimes, payment transactions are repeated by accident; it could be due to user error, API failure or a retry error that causes a credit card to be charged twice.
Using the transactions table, identify any payments made at the same merchant with the same credit card for the same amount within 10 minutes of each other. Count such repeated payments.

Assumptions:

- The first transaction of such payments should not be counted as a repeated payment. This means, if there are two transactions performed by a merchant with the same credit card and for the same amount within 10 minutes, there will only be 1 repeated payment.

`transactions` Table:

Column Name	Type
transaction_id	integer
merchant_id	integer
credit_card_id	integer
amount	integer
transaction_timestamp	datetime

`transactions` Example Input:

transaction_id	merchant_id	credit_card_id	amount	transaction_timestamp
1	101	1	100	09/25/2022 12:00:00
2	101	1	100	09/25/2022 12:08:00
3	101	1	100	09/25/2022 12:28:00
4	102	2	300	09/25/2022 12:00:00
6	102	2	400	09/25/2022 14:00:00

Example Output:

payment_count
1