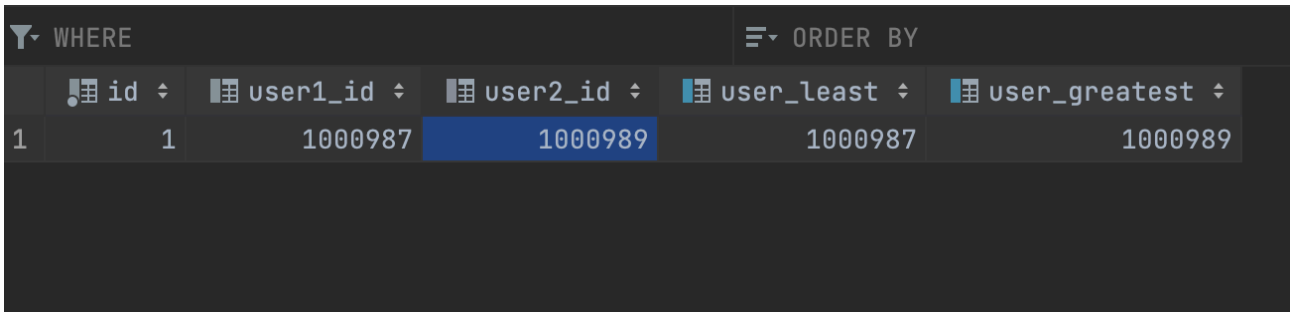
Для выполнения этой задачи были внесены изменения в основной сервис (https://github.com/plohoicod/otus_highload_5.1), была добавлена таблица диалогов, REST методы для создания сообщений и REST методы для межсервисного взаимодействия.

Перед отправкой сообщения создается диалог с уникальным id для двух пользователей:



WHERE ORDER BY

	id	user1_id	user2_id	user_least	user_greatest
1	1	1000987	1000989	1000987	1000989

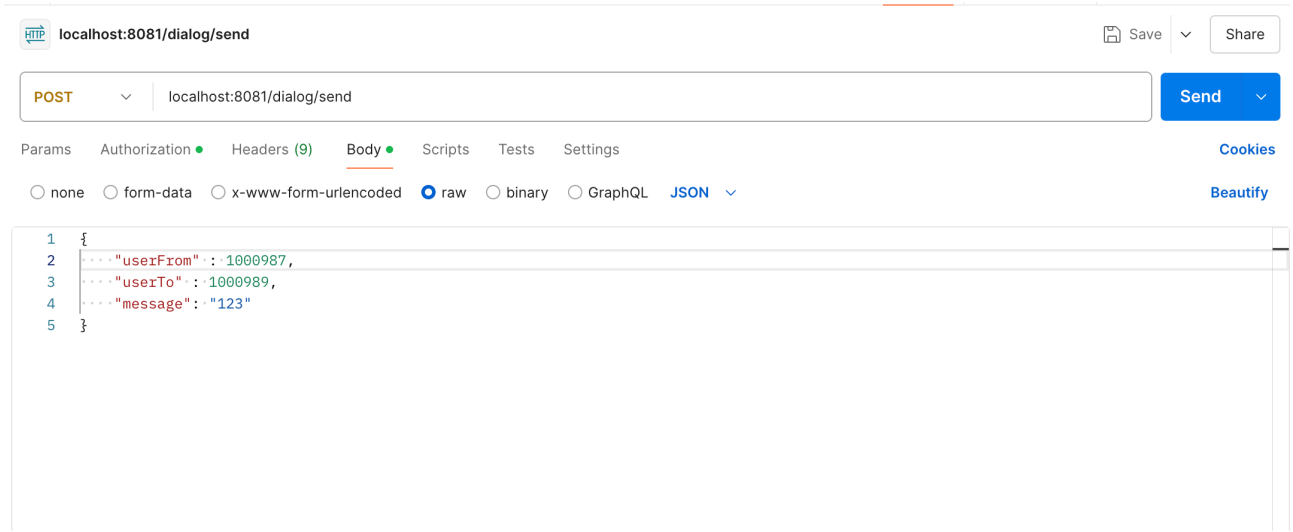
Далее этот найди будет использоваться для шардирования сообщений в следующем микросервисе - https://github.com/plohoicod/otus_highload_5.2 .

Алгоритм шардирования:
1. Считаем остаток от деления id диалога на 64(число выбрано слуайно, можно выбрать и другое, подходящее под конкретную задачу). Так же можно посчитать хеш функцию, и после этого от нее взять остаток от деления(если ключ текстовый или составной к примеру)
2. У каждого шарда есть диапазон значений, в один из которых попадает результат.
3. Записываем результат в соответствующий шард
4. Получаем результат так-же, через остаток от деления и выбор шарда по диапазону

Данный алгоритм не требует перемещения всех элементов между шарадами, если изменяется кол-во шардов. В текущей реализации конфцигурация двух шардов сделана в приложении, в дальнейшем ее следует перенести в базу данных.

Автоматический решардинг не предусмотрен, но можно последовать следующему алгоритму:
Пишем данные всегда в новые шарды. Читаем сначала из нового шарда; если там нет нужных данных, то обращаемся к старым шардам. Если данные оказываются в старом шарде, то при желании можно их переносить в новый — и в конце концов данные перераспределятся между шардами. Так же можно параллельно запустить процесс автоматический перенос данных, к тем к которым не идут запросы.

Создаем сообщение в диалоге:



Оно появилось в том же шарде, что и прошлое сообщение:

WHERE ORDER BY

	id	chat_id	user_from_id	user_to_id	message	create_datetime
1	3	1	1000989	1000987	123	2024-08-02 11:11:39.006946
2	4	1	1000987	1000989	123	2024-08-02 11:33:39.023552

Получаем сообщения диалога, они отсортированы по времени: