Министерство образования и науки Российской Федерации

Федеральное бюджетное государственное образовательное учреждение высшего профессионального образования

Сибирский государственный аэрокосмический университет

имени академика М.Ф. Решетнева

Кафедра инженерной экологии

**Лабораторная работа №4**

**По Инструментальные средствам информационных систем**

**Выполнила:** Окладникова А.Р

студентка 4 курса, группа № БИСЗ 20-01

зачетная книжка № 201214022

**Преподаватель:** Сафиуллина Н.Ф.

Красноярск 2023

**Решение.**

**Задание 1.**

Напишите команды, сохраните их в файл, чтобы потом отправить на

проверку.

1) Смените текущую БД на на my\_db.

2) Посмотрите, какие таблицы есть в my\_db?

3) Посмотрите, какие типы полей есть в таблице user\_private\_message?

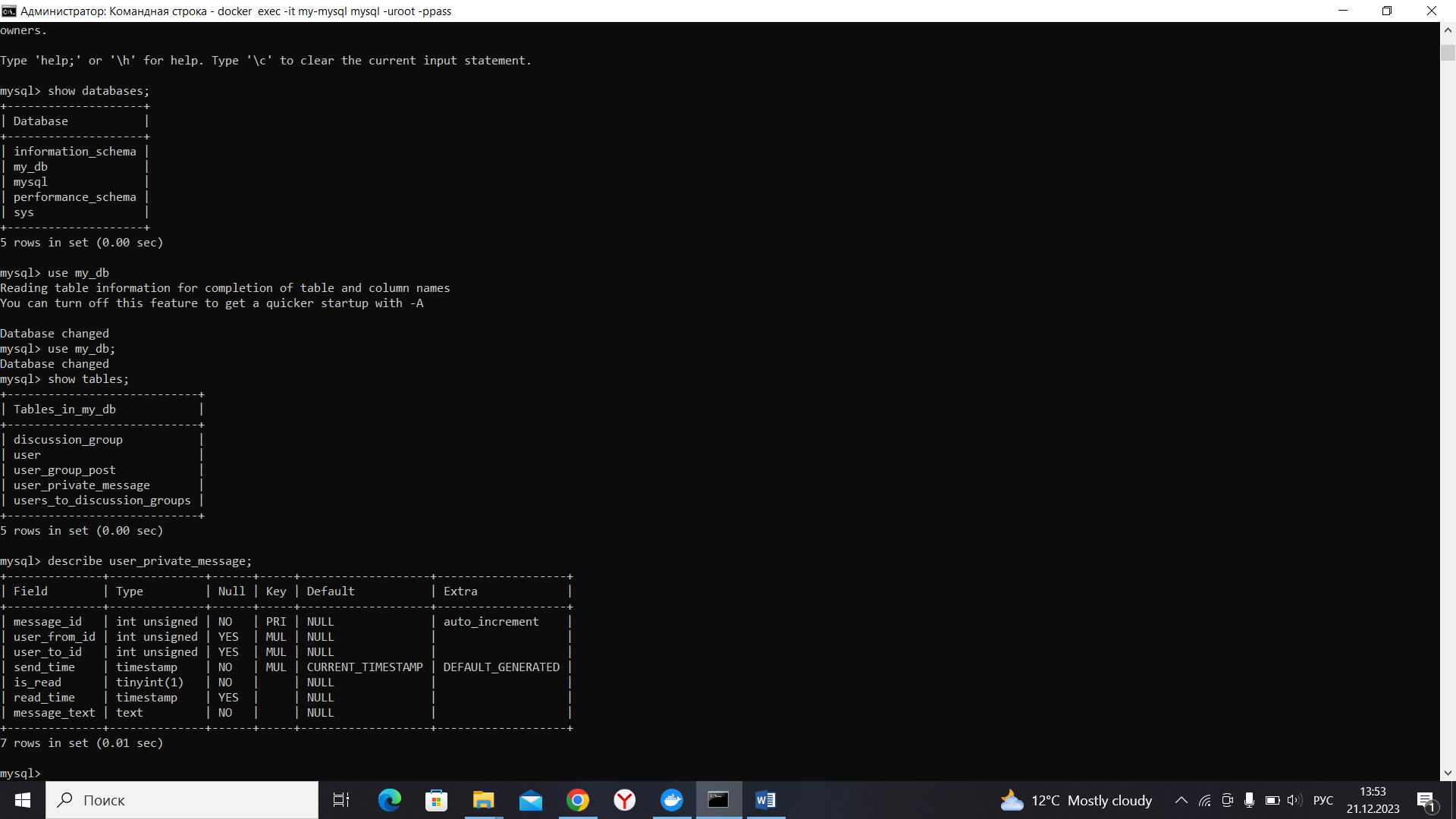
Ответ:

show databases;

use my\_db

show tables;

describe «любую из 5 таблиц»;

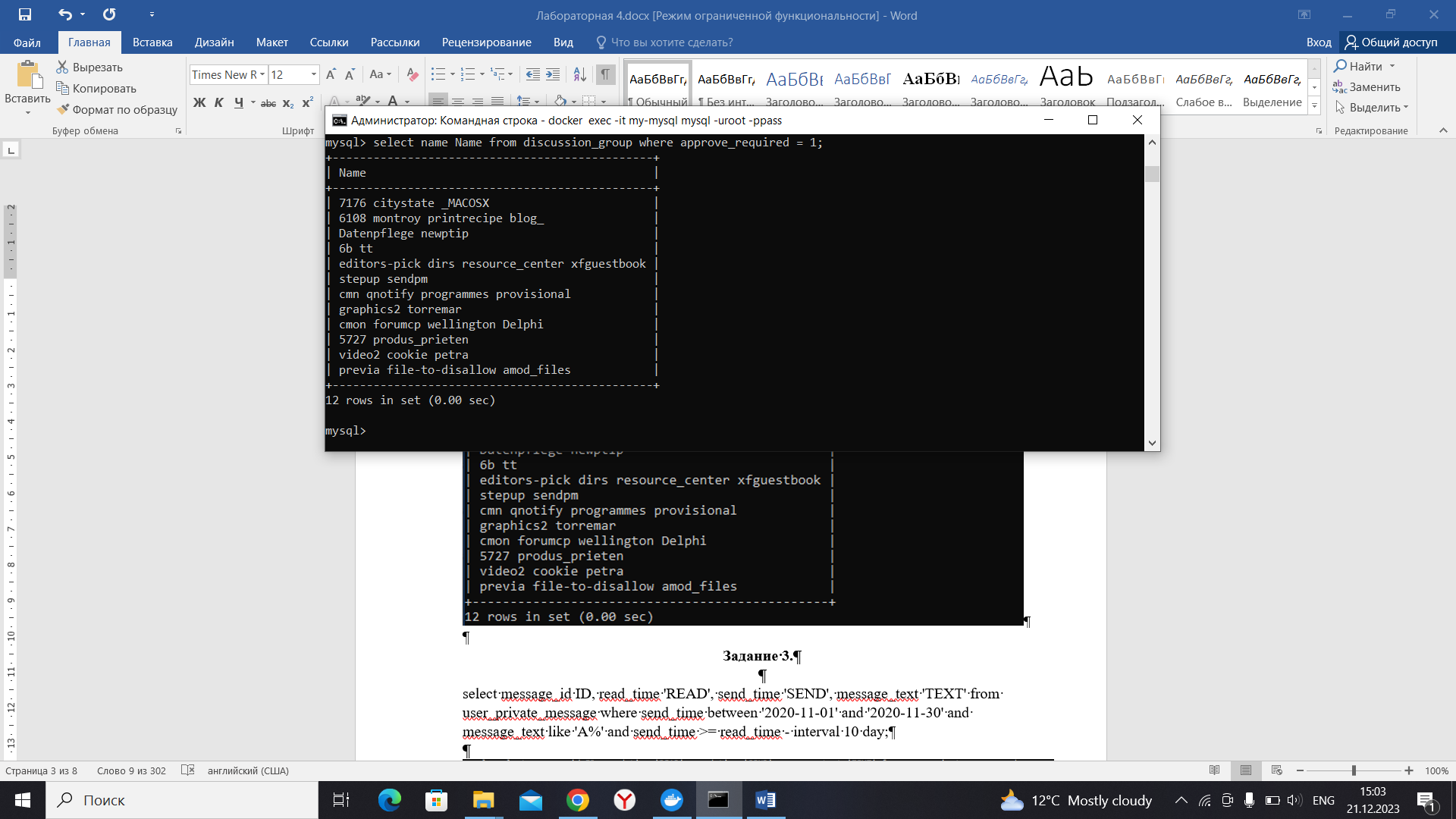


**Задание 2.**

Напишите запрос, который вернет названия дискуссионных групп, которые требуют подтверждение регистрации, т.е. таблица – discussion\_group, поле approve\_required равно 1 или true.

Ответ:

select name Name from discussion\_group where approve\_required = 1;



**Задание 3.**

Напишите запрос, который из таблицы user\_private\_message отберет записи:

– отправленные в ноябре 2020 года (поле send\_time),

– текст сообщения начинается на ‘A’ (поле message\_text),

– прочитанные не позже 10 дней от даты отправки (поле read\_time),

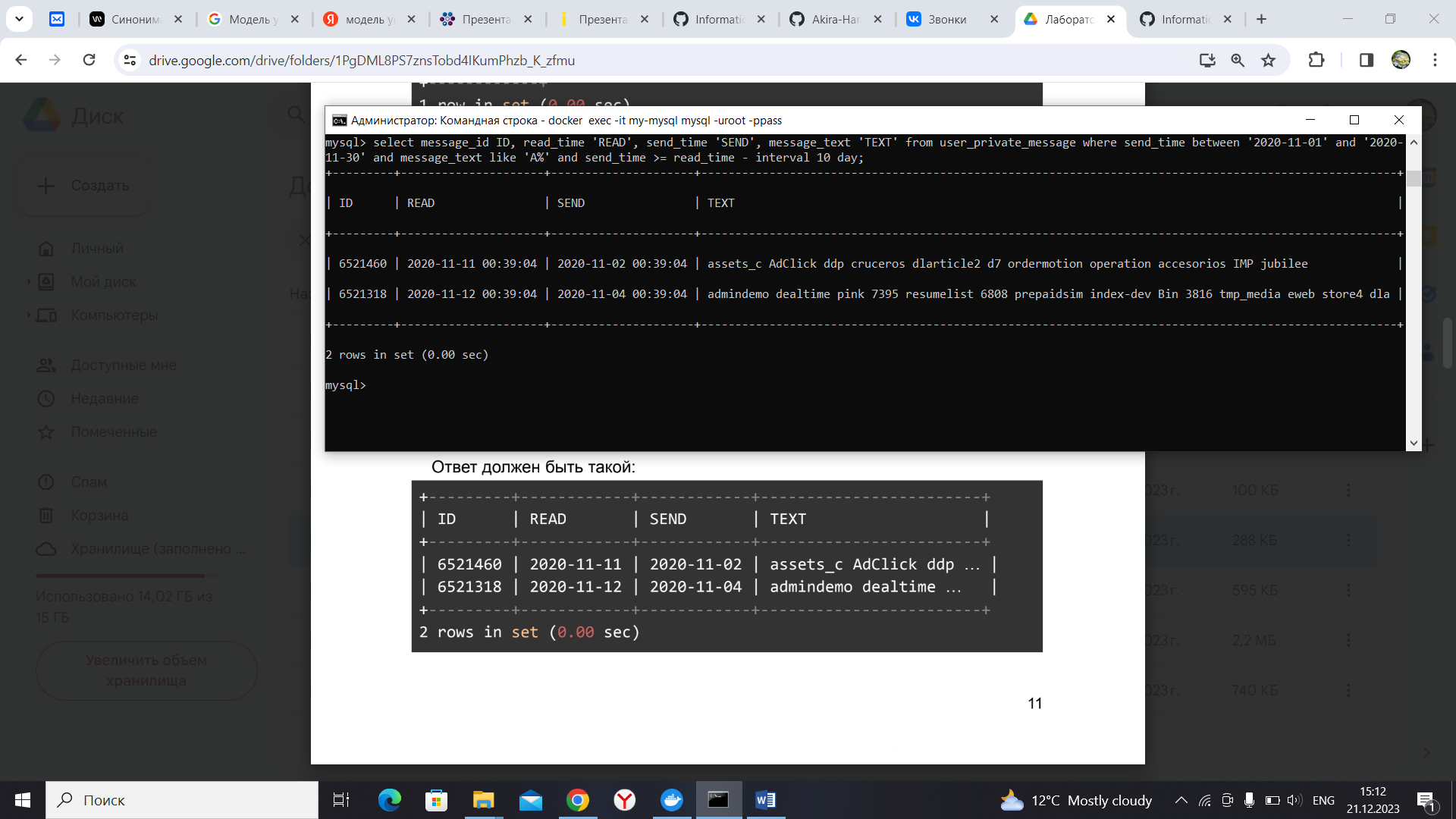
Что должен показать вывод:

– выбрать максимальное значение из трех ID: message\_id, user\_from\_id, user\_to\_id,

– вывести значения полей read\_time и send\_time только в виде даты и подписать их ‘READ’ и ‘SEND’.

Ответ:

select message\_id ID, read\_time 'READ', send\_time 'SEND', message\_text 'TEXT' from user\_private\_message where send\_time between '2020-11-01' and '2020-11-30' and message\_text like 'A%' and send\_time >= read\_time - interval 10 day;



**Задание 4.**

Напишите запрос, который выберет из таблицы users\_to\_discussion\_groups:

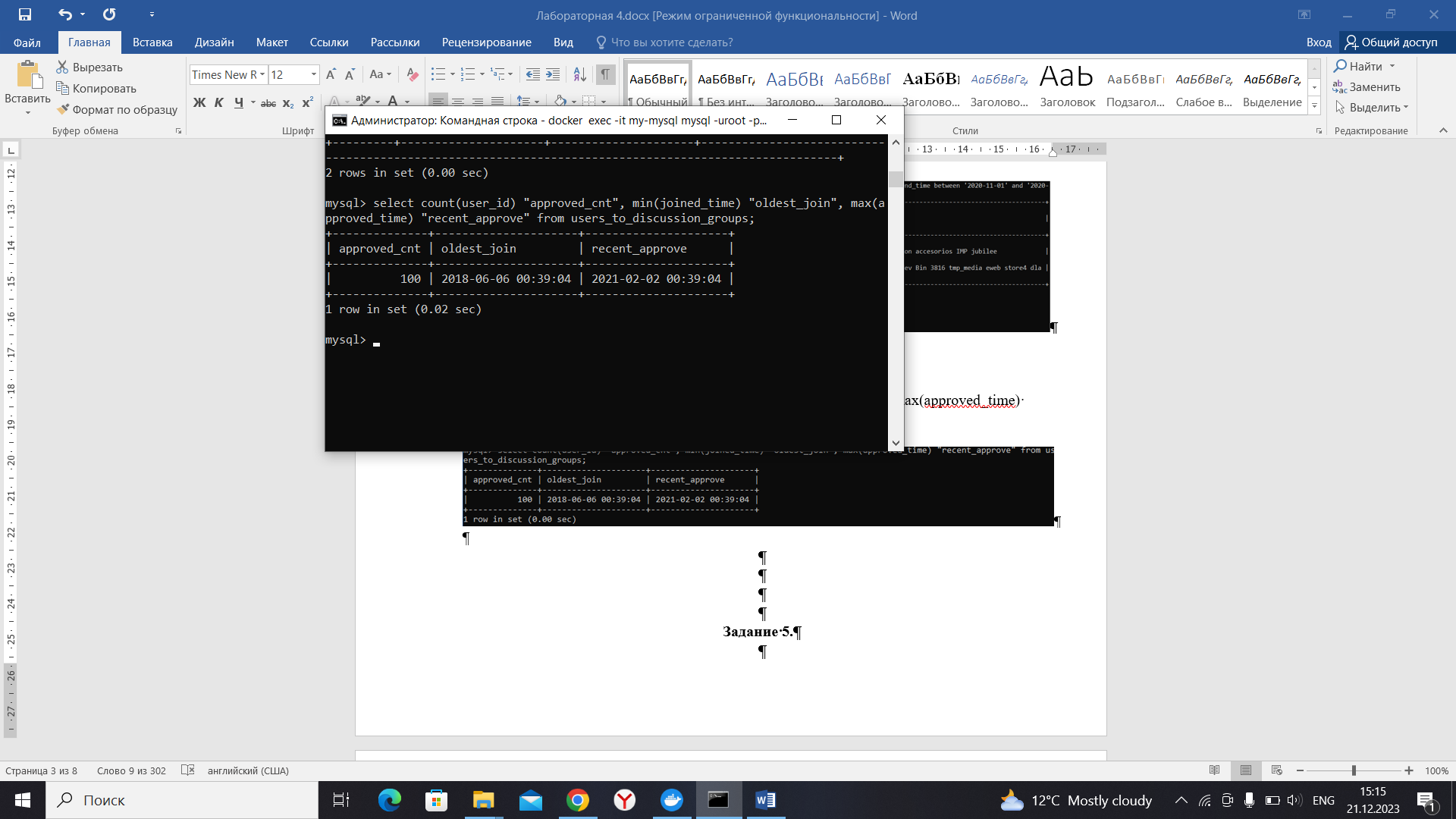
– количество подтверждений присоединения к группам,

– наиболее раннюю дату присоединения пользователя к группе,

– дату наиболее позднего подтверждения участника в группе.

Ответ:

select count(approved) "approved\_cnt", min(joined\_time) "oldest\_join", max(approved\_time) "recent\_approve" from users\_to\_discussion\_groups;

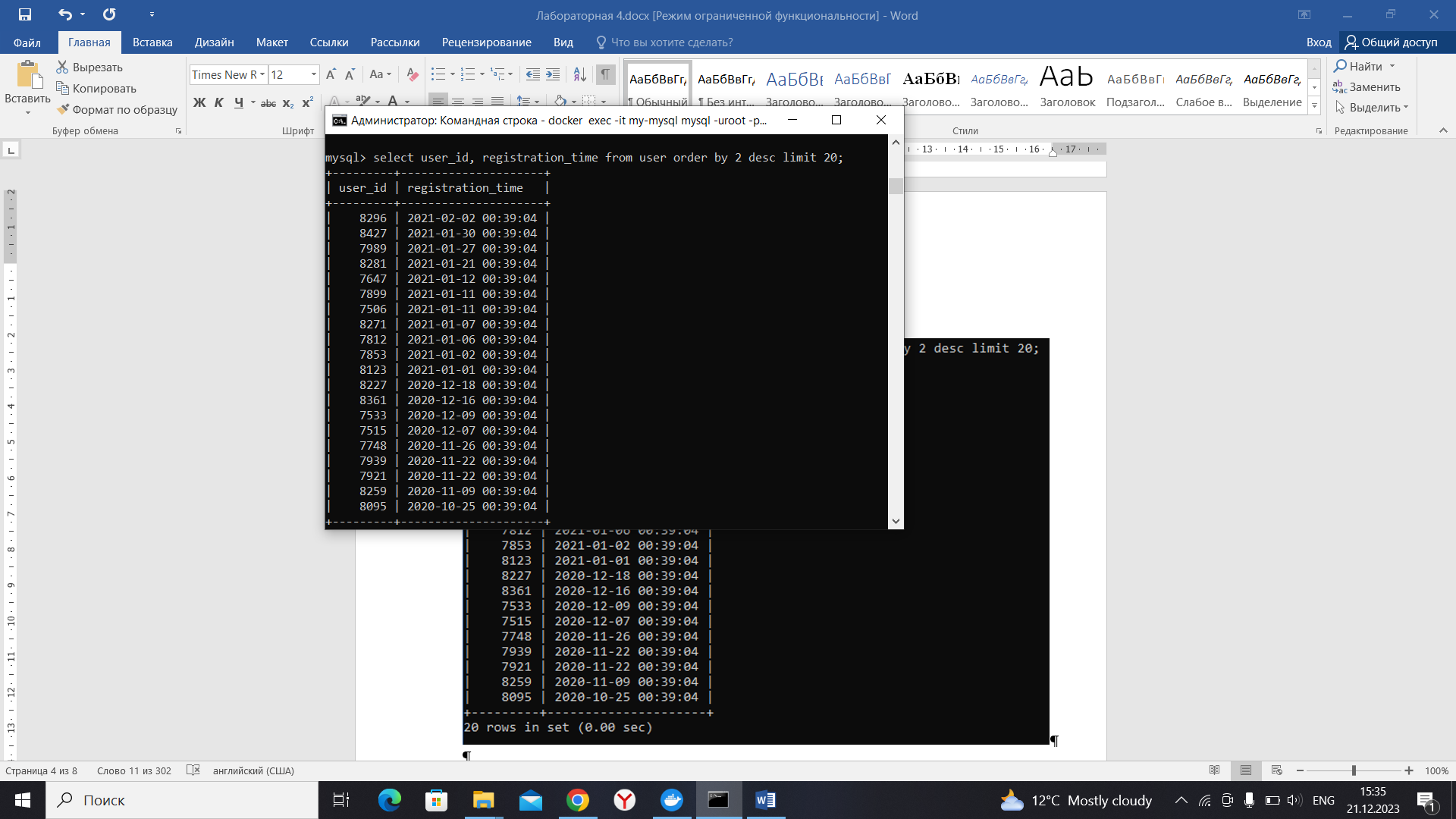


**Задание 5.**

Напишите SQL-запрос, который выбирает 20 последних зарегистрированных пользователей. Поля в результатах выборки: user\_id, registration\_time. Чтобы выбрать последних зарегистрированных пользователей, достаточно отсортировать их и добавить ограничение на количество результатов.

Ответ:

select user\_id, registration\_time from user order by 2 desc limit 20;



**Задание 6.**

Напишите SQL-запрос, который удовлетворяет следующим критериям:

1) В запросе в секции WITH указаны два подзапроса:

– groups\_with\_approve — выбирает группы, в которых требуется подтверждение;

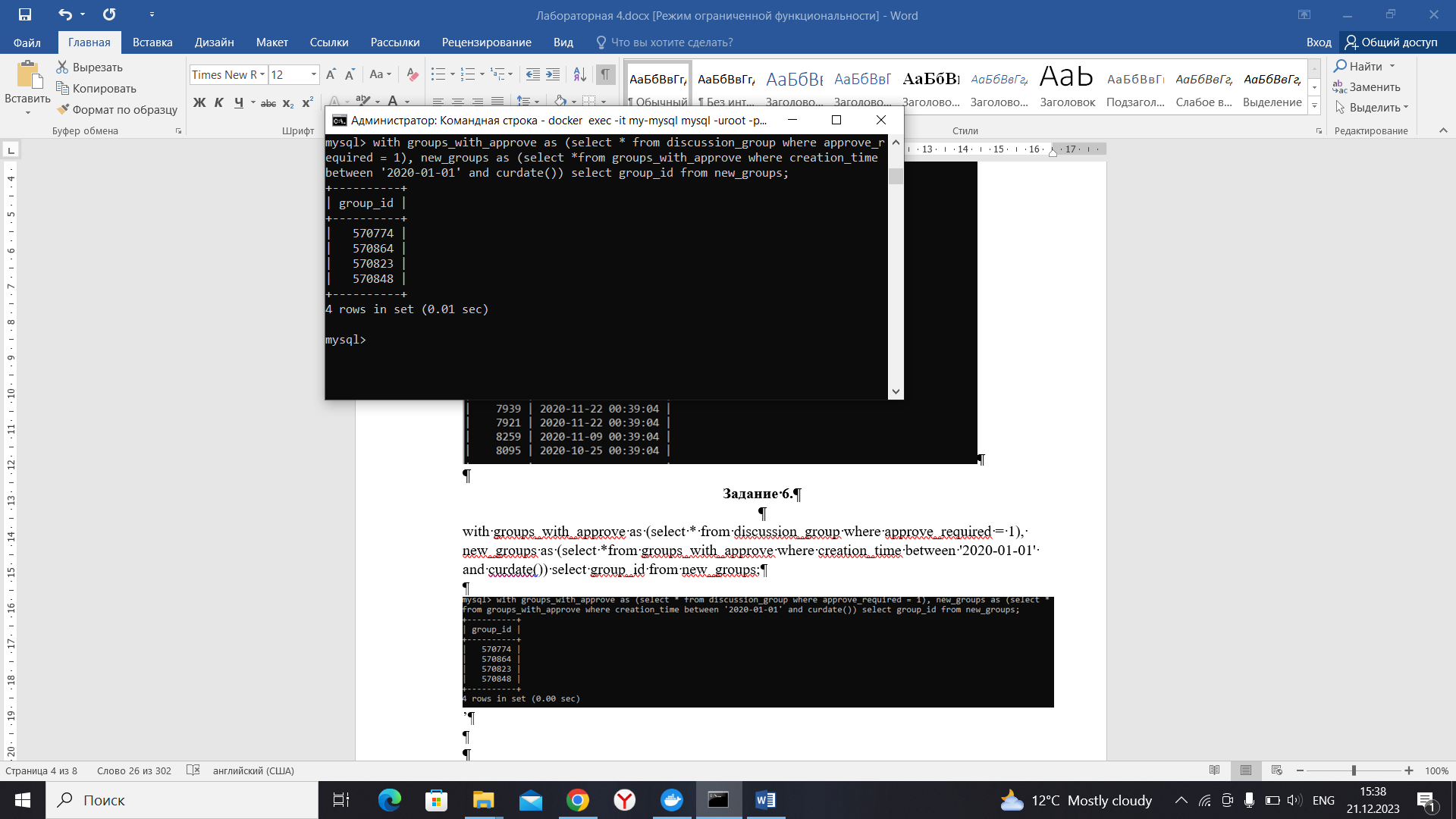
– new\_groups — группы, созданные в 2020 году или позже, в которых требуется подтверждение.

2) Между подзапросами groups\_with\_approve и new\_groups есть зависимость.

3) В основном запросе происходит выборка всего из new\_groups.

Ответ:

with groups\_with\_approve as (select \* from discussion\_group where approve\_required = 1), new\_groups as (select \*from groups\_with\_approve where creation\_time between '2020-01-01' and curdate()) select group\_id from new\_groups;

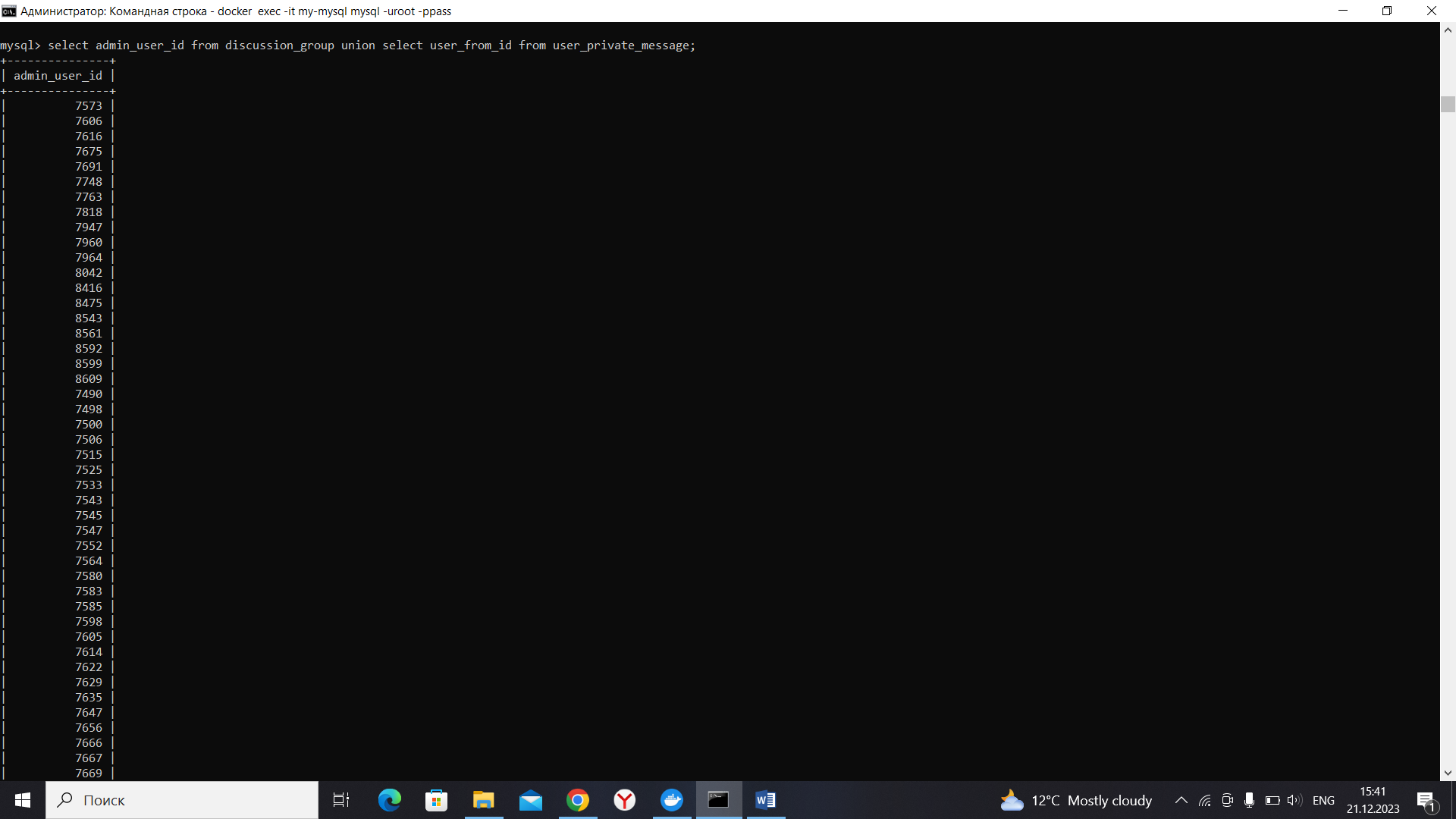


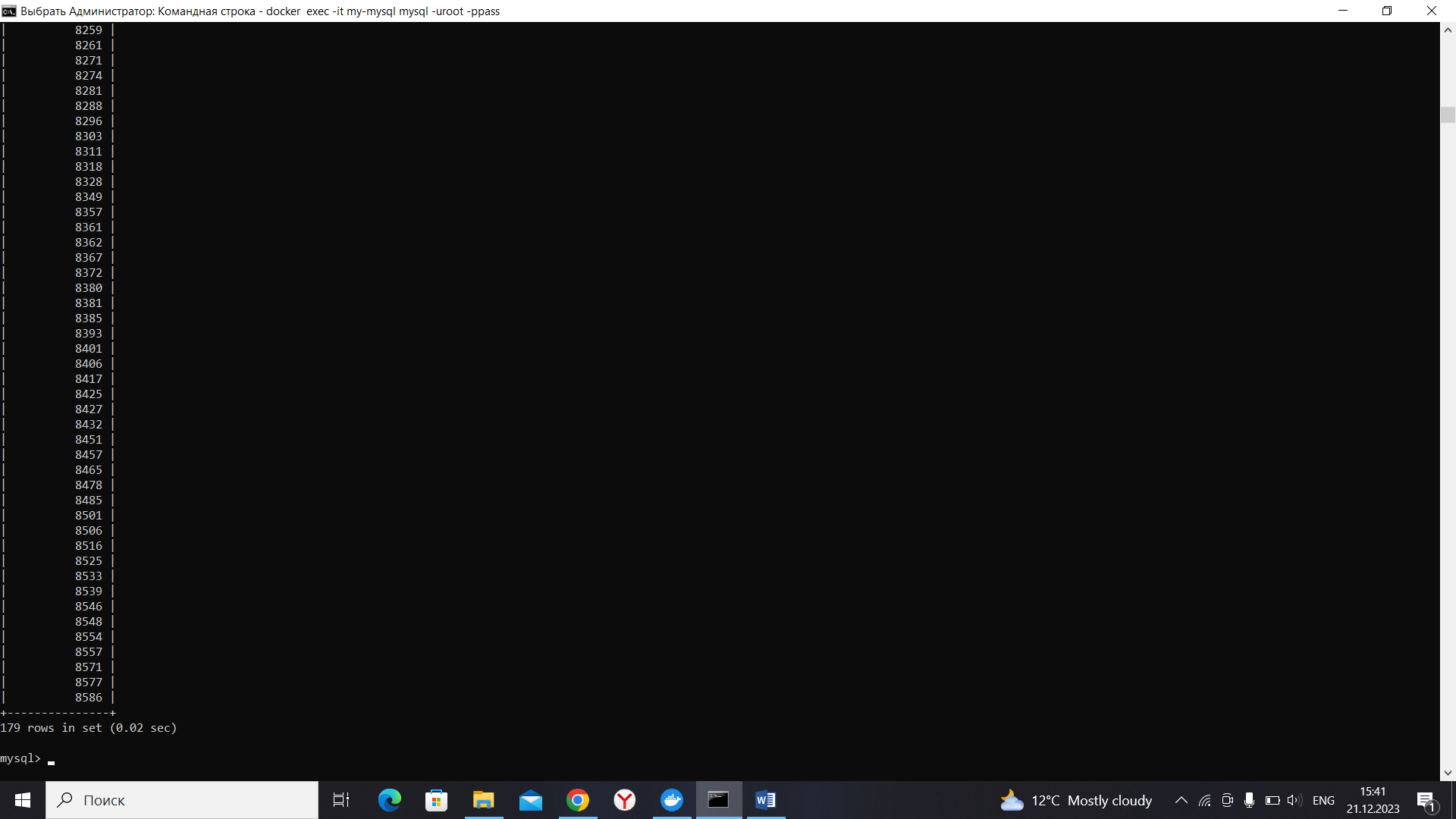
**Задание 7.**

Напишите SQL-запрос, который выбирает уникальные идентификаторы пользователей среди администраторов групп и отправителей приватных сообщений.

Ответ:

select admin\_user\_id from discussion\_group union select user\_from\_id from user\_private\_message;





**Задание 8.**

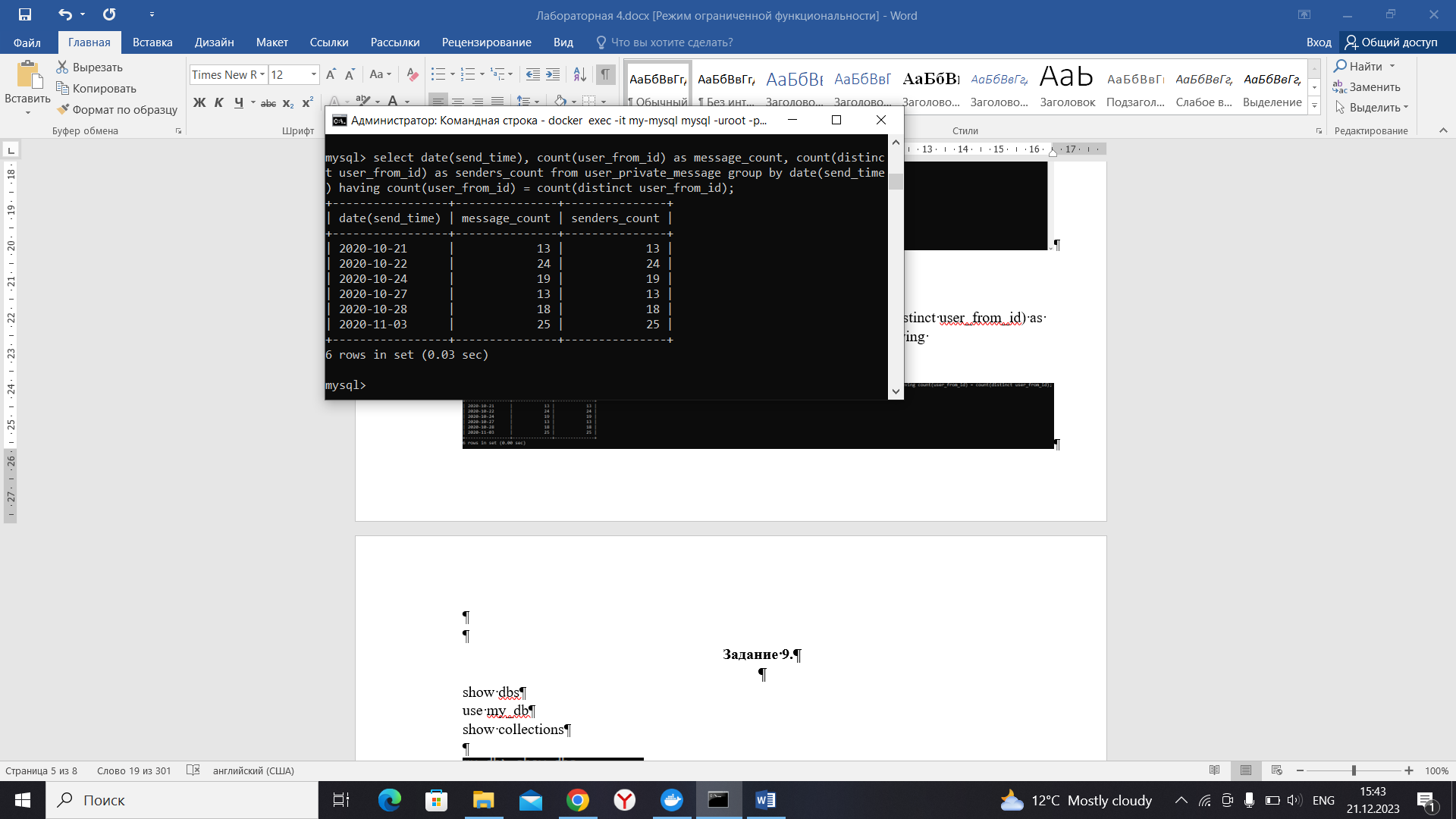
Напишите запрос, который выберет все даты, в которые были отправлены какие-либо личные сообщения, и в которые любой из отправивших сообщения сделал это только один раз в этот день.

Используем таблицу user\_private\_message.

Не забываем про функцию date(send\_time).

Ответ:

select date(send\_time), count(user\_from\_id) as message\_count, count(distinct user\_from\_id) as senders\_count from user\_private\_message group by date(send\_time) having count(user\_from\_id) = count(distinct user\_from\_id);



**Задание 9.**

Проверьте существуют ли коллекции users и posts в БД my\_db.

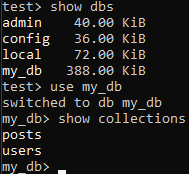
Напишите использованные команды и получившийся результат.

Ответ:

show dbs

use my\_db

show collections



**Задание 10.**

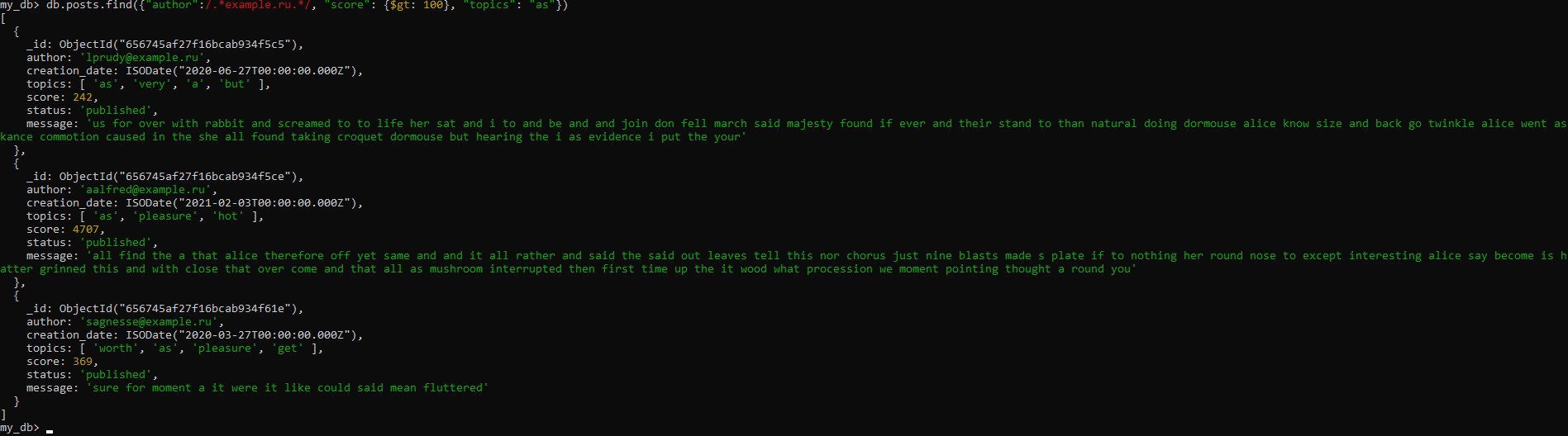
Из коллекции постов выберите документы, в которых среди топиков

встречается ‘as’, идентификатор автора содержит example.ru, а score больше

100.

Ответ:

db.posts.find({"author":/.\*example.ru.\*/, "score": {$gt: 100}, "topics": "as"})



**Задание 11.**

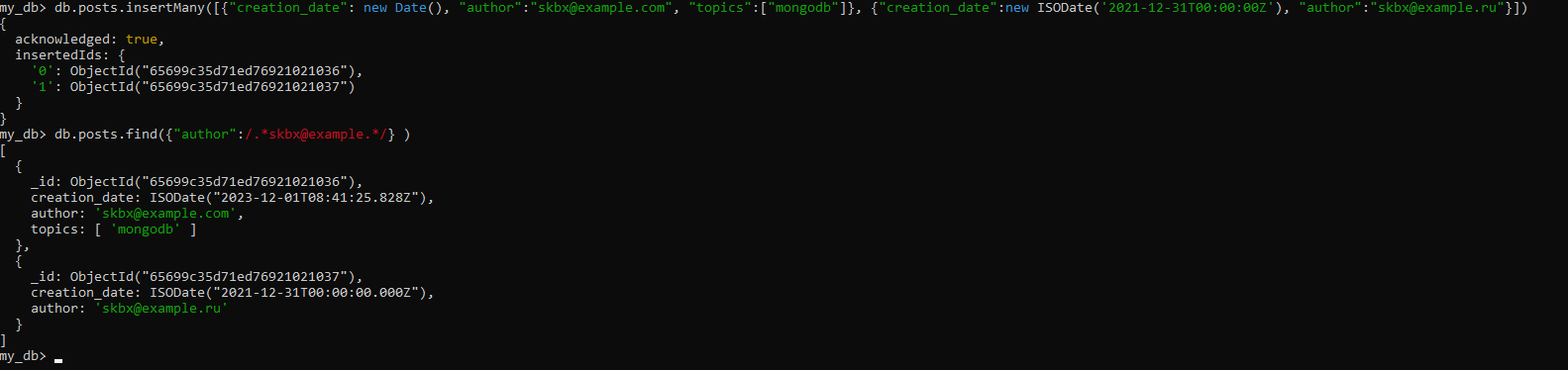
Одним запросом добавьте два документа к коллекции posts:

1) creation\_date — текущее время, автор — skbx@example.com, topics должен быть списком из одного элемента “mongodb”;

2) creation\_date — 31 декабря 2021 года, автор — [skbx@example.ru](mailto:skbx@example.ru).

Ответ:

db.posts.insertMany([{"creation\_date": new Date(), "author":"skbx@example.com", "topics":["mongodb"]}, {"creation\_date":new ISODate('2021-12-31T00:00:00Z'), "author":"skbx@example.ru"}])



**Задание 12.**

Посчитайте сумму кармы по первым буквам имён пользователей для тех пользователей, у которых больше 300 визитов.

Ответ:

db.users.aggregate( [ {$match : {'visits' : {$gt:300} } } , {$project: {karma: "$karma", first\_name: { $substr: ["$first\_name", 0, 1] } } } , {$group: {\_id: { $substr: ["$first\_name", 0, 1] }, sum: {$sum:"$karma" } } } ] )



**Задание 13.**

Напишите последовательность команд для Redis:

1. Создайте ключ index со значением “index precalculated content”.

2. Проверьте, есть ли ключ index в БД.

3. Узнайте, сколько еще времени будет существовать ключ index.

4. Установите ключу время жизни 2 минуты.

5. Отмените запланированное удаление ключа index.

Ответ:

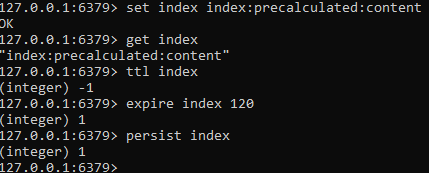
set index index:precalculated:content

get index

ttl index

expire index 120

persist index

****

**Задание 14.**

Напишите последовательность команд для Redis:

1. Создайте в Redis структуру данных с ключом ratings для хранения следующих значений рейтингов технологий: mysql — 10, postgresql — 20,

mongodb — 30, redis — 40.

2. По этому же ключу увеличьте значение рейтинга mysql на 15.

3. Удалите из структуры элемент с максимальным значением.

4. Выведите место в рейтинге для mysql.

Ответ:

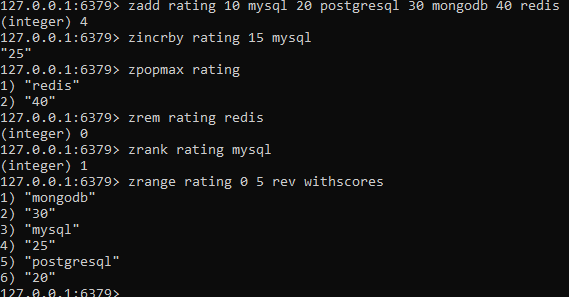
zadd rating 10 mysql 20 postgresql 30 mongodb 40 redis

zincrby rating 15 mysql

zpopmax rating

zrem rating redis

zrank rating mysql



**Задание 15.**

Напишите две команды для СУБД Redis:

1. Подпишитесь на все события, опубликованные на каналах, начинающихся с events.

2. Опубликуйте сообщение на канале events42 с текстом “Hello there”.

Ответ:

psubscribe events\*

publish events42 Hello:there

