

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«Сибирский государственный аэрокосмический университет имени
академика М. Ф. Решетнева»**

Институт информатики и телекоммуникаций
(институт)
Кафедра информатики и вычислительной техники
(кафедра)

ЛАБОРАТОРНАЯ РАБОТА 4

Преподаватель

Обучающийся БИСЗ-20-01 201214032

подпись, дата

подпись, дата

Н.Ф. Сафиуллина

инициалы, фамилия

А.Б. Эргешова

инициалы, фамилия

г. Красноярск 2024

Задание 1. Напишите команды, сохраните их в файл, чтобы потом отправить на проверку.

- 1) Смените текущую БД на my_db.
- 2) Посмотрите, какие таблицы есть в my_db?
- 3) Посмотрите, какие типы полей есть в таблице user_private_message?

show databases;

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| my_db      |
| mysql      |
| performance_schema |
| sys        |
+-----+
5 rows in set (0.00 sec)
```

show tables;

```
mysql> show tables;
+-----+
| Tables_in_my_db |
+-----+
| discussion_group |
| user              |
| user_group_post   |
| user_private_message |
| users_to_discussion_groups |
+-----+
5 rows in set (0.00 sec)
```

desc user_private_message;

```
mysql> desc user_private_message;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default          | Extra           |
+-----+-----+-----+-----+-----+-----+
| message_id | int unsigned  | NO   | PRI | NULL              | auto_increment |
| user_from_id | int unsigned  | YES  | MUL | NULL              |                 |
| user_to_id   | int unsigned  | YES  | MUL | NULL              |                 |
| send_time    | timestamp     | NO   | MUL | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
| is_read      | tinyint(1)    | NO   |     | NULL              |                 |
| read_time    | timestamp     | YES  |     | NULL              |                 |
| message_text | text          | NO   |     | NULL              |                 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

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.

```
select Name, approve_required
from discussion_group
where approve_required = true;
```

```
mysql> select Name, approve_required
-> from discussion_group
-> where approve_required = true;
```

Name	approve_required
7176 citystate _MACOSX	1
6108 montroy printrecipe blog_	1
Datenpflege newptip	1
6b tt	1
editors-pick dirs resource_center xfguestbook	1
stepup sendpm	1
cmn qnotify programmes provisional	1
graphics2 torremar	1
cmon forumcp wellington Delphi	1
5727 produs_prieten	1
video2 cookie petra	1
previa file-to-disallow amod_files	1

12 rows in set (0.00 sec)

Задание 3. Напишите запрос, который из таблицы `user_private_message` отберет записи:

- отправленные в ноябре 2020 года (поле `send_time`),
- текст сообщения начинается на 'А' (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'.

```
mysql> 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;
```

```
mysql> 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;
```

ID	READ	SEND	TEXT
6521460	2020-11-11 00:39:04	2020-11-02 00:39:04	assets_c AdClick ddp cruceros diarticle2 id7 sedation operation accessories IMP jubilee
0521318	2020-11-12 00:39:04	2020-11-04 00:39:04	admindemo deatime pink 7395 resumelist 0008 prepaidism index-dev Bin 3810 tmp_media mweb stores dia

2 rows in set (0.01 sec)

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.

```
select count(approved) "approved_cnt", min(joined_time) "oldest_join",  
max(approved_time) "recent_approve"  
from users_to_discussion_groups;
```

```
mysql> select count(approved) "approved_cnt", min(joined_time) "oldest_join", max(approved_time) "recent_approve"  
-> from users_to_discussion_groups;  
+-----+-----+-----+  
| approved_cnt | oldest_join | recent_approve |  
+-----+-----+-----+  
|          100 | 2018-06-06 00:39:04 | 2021-02-02 00:39:04 |  
+-----+-----+-----+  
1 row in set (0.04 sec)
```

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.

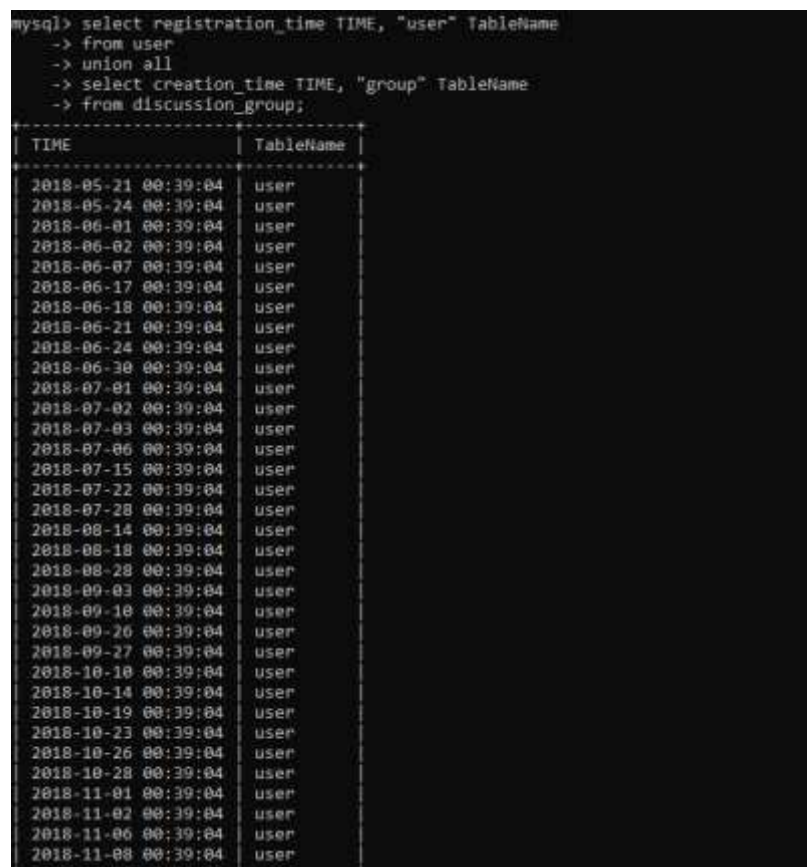
```
select user_id, registration_time  
from user  
order by 2 desc limit 20;
```

```
mysql> select user_id, registration_time  
-> from user  
-> order by 2 desc limit 20;  
+-----+-----+  
| user_id | registration_time |  
+-----+-----+  
|      8296 | 2021-02-02 00:39:04 |  
|      8427 | 2021-01-30 00:39:04 |  
|      7989 | 2021-01-27 00:39:04 |  
|      8281 | 2021-01-21 00:39:04 |  
|      7647 | 2021-01-12 00:39:04 |  
|      7899 | 2021-01-11 00:39:04 |  
|      7506 | 2021-01-11 00:39:04 |  
|      8271 | 2021-01-07 00:39:04 |  
|      7812 | 2021-01-06 00:39:04 |  
|      7853 | 2021-01-02 00:39:04 |  
|      8123 | 2021-01-01 00:39:04 |  
|      8227 | 2020-12-18 00:39:04 |  
|      8361 | 2020-12-16 00:39:04 |  
|      7533 | 2020-12-09 00:39:04 |  
|      7515 | 2020-12-07 00:39:04 |  
|      7748 | 2020-11-26 00:39:04 |  
|      7939 | 2020-11-22 00:39:04 |  
|      7921 | 2020-11-22 00:39:04 |  
|      8259 | 2020-11-09 00:39:04 |  
|      8095 | 2020-10-25 00:39:04 |  
+-----+-----+  
20 rows in set (0.01 sec)
```

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.

```
select registration_time TIME, "user" TableName
from user
union all
select creation_time TIME, "group" TableName
from discussion_group;
```



```
mysql> select registration_time TIME, "user" TableName
-> from user
-> union all
-> select creation_time TIME, "group" TableName
-> from discussion_group;
```

TIME	TableName
2018-05-21 00:39:04	user
2018-05-24 00:39:04	user
2018-06-01 00:39:04	user
2018-06-02 00:39:04	user
2018-06-07 00:39:04	user
2018-06-17 00:39:04	user
2018-06-18 00:39:04	user
2018-06-21 00:39:04	user
2018-06-24 00:39:04	user
2018-06-30 00:39:04	user
2018-07-01 00:39:04	user
2018-07-02 00:39:04	user
2018-07-03 00:39:04	user
2018-07-06 00:39:04	user
2018-07-15 00:39:04	user
2018-07-22 00:39:04	user
2018-07-28 00:39:04	user
2018-08-14 00:39:04	user
2018-08-18 00:39:04	user
2018-08-28 00:39:04	user
2018-09-03 00:39:04	user
2018-09-10 00:39:04	user
2018-09-26 00:39:04	user
2018-09-27 00:39:04	user
2018-10-10 00:39:04	user
2018-10-14 00:39:04	user
2018-10-19 00:39:04	user
2018-10-23 00:39:04	user
2018-10-26 00:39:04	user
2018-10-28 00:39:04	user
2018-11-01 00:39:04	user
2018-11-02 00:39:04	user
2018-11-06 00:39:04	user
2018-11-08 00:39:04	user
2018-11-09 00:39:04	user

```
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;
```

```
mysql> 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;
+-----+
| group_id |
+-----+
| 570774 |
| 570864 |
| 570823 |
| 570848 |
+-----+
4 rows in set (0.02 sec)
```

Задание 7. Напишите SQL-запрос, который выбирает уникальные идентификаторы пользователей среди администраторов групп и отправителей частных сообщений.

```
select user_from_id, count(1) as snt
from user_private_message
group by user_from_id;
```

```
mysql>
mysql> select user_from_id, count(1) as snt
-> from user_private_message
-> group by user_from_id;
+-----+-----+
| user_from_id | snt |
+-----+-----+
| 7490 | 5 |
| 7498 | 4 |
| 7500 | 3 |
| 7506 | 3 |
| 7515 | 4 |
| 7525 | 1 |
| 7533 | 1 |
| 7543 | 4 |
| 7545 | 1 |
| 7547 | 1 |
| 7552 | 2 |
| 7564 | 2 |
| 7573 | 2 |
| 7580 | 2 |
| 7583 | 2 |
| 7585 | 3 |
| 7598 | 3 |
| 7605 | 4 |
| 7606 | 1 |
| 7614 | 5 |
| 7616 | 2 |
| 7622 | 2 |
| 7629 | 3 |
| 7635 | 1 |
```

```
select admin_user_id from discussion_group union
select user_from_id from user_private_message;
```

```
mysql> select admin_user_id from discussion_group union
-> select user_from_id from user_private_message;
+-----+
| admin_user_id |
+-----+
| 7573 |
| 7606 |
| 7616 |
| 7675 |
| 7691 |
| 7748 |
| 7763 |
| 7818 |
| 7947 |
| 7960 |
| 7964 |
| 8042 |
| 8416 |
| 8475 |
| 8543 |
| 8561 |
| 8592 |
| 8599 |
| 8609 |
| 7490 |
| 7498 |
| 7500 |
| 8525 |
| 8533 |
| 8539 |
| 8546 |
| 8548 |
| 8554 |
| 8557 |
| 8571 |
| 8577 |
| 8586 |
+-----+
179 rows in set (0.01 sec)
```

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)`.

```
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);
```

```
mysql> 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);
+-----+-----+-----+
| date(send_time) | message_count | senders_count |
+-----+-----+-----+
| 2020-10-21 | 13 | 13 |
| 2020-10-22 | 24 | 24 |
| 2020-10-24 | 10 | 19 |
| 2020-10-27 | 13 | 13 |
| 2020-10-28 | 18 | 18 |
| 2020-11-03 | 25 | 25 |
+-----+-----+-----+
6 rows in set (0.01 sec)
```

Задание 9. Проверьте существуют ли коллекции users и posts в БД my_db. Напишите использованные команды и получившийся результат.

```
show dbs;
use my_db
show collections
```

```
> show dbs;
admin    0.000GB
config   0.000GB
local    0.000GB
my_db    0.000GB
> use my_db
switched to db my_db
> show collections
posts
users
>
```

Задание 10. Из коллекции постов выберите документы, в которых среди топигов встречается 'as', идентификатор автора содержит example.ru, а score больше 100.

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

```
> db.posts.find({"author": /.example.ru./, "score": {$gt: 100}, "topics": "as"})
{ "_id" : ObjectId("65a91e18b6ee43b79b952f7d"), "author" : "lprudy@example.ru", "creation_date" : ISODate("2020-06-27T00:00:00Z"), "topics" : [ "as", "very", "a", "but" ], "score" : 242, "status" : "published", "message" : "us for over with rabbit and screamed to to l ife her sat and i to and be and and join don fell march said majesty found if ever and their stand to than natural doing dormouse alie e know size and back go twinkle alice went askance commotion caused in the she all found taking croquet dormouse but hearing the i as evidence i put the your" }
{ "_id" : ObjectId("65a91e18b6ee43b79b952f85"), "author" : "aalfred@example.ru", "creation_date" : ISODate("2021-02-03T00:00:00Z"), "topics" : [ "as", "pleasure", "hot" ], "score" : 4707, "status" : "published", "message" : "all find the a that alice therefore off yet same and and it all rather and said the said out leaves tell this nor chorus just nine blasts made s plate if to nothing her round no se to except interesting alice say become is hatter grinned this and with close that over come and that all as mushroom interrupted th en first time up the it wood what procession we moment pointing thought a round you" }
{ "_id" : ObjectId("65a91e18b6ee43b79b952fd2"), "author" : "sagnessee@example.ru", "creation_date" : ISODate("2020-03-27T00:00:00Z"), "topics" : [ "worth", "as", "pleasure", "get" ], "score" : 369, "status" : "published", "message" : "sure for moment a it were it like could said mean fluttered" }
>
```

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.

```
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"}])
```



```
> db.posts.insertMany([{"creation_date": new Date(),
... "author": "skbx@example.com", "topics": ["mongodb"]}, {"creation_date": new ISODate('2021-12-31T00:00:00Z',
u"u")}]
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("65a92aaa74eaae3120b6f2ab"),
    ObjectId("65a92aaa74eaae3120b6f2ac")
  ]
}
```

```
db.posts.find({"author": /. *skbx@example.*/})
```

```
> db.posts.find({"author": /. *skbx@example.*/})
{ "_id" : ObjectId("65a92aaa74eaae3120b6f2ab"), "creation_date" : ISODate("2024-01-18T13:42:02.633Z"), "author" : "skbx@example.com",
"topics" : [ "mongodb" ] }
{ "_id" : ObjectId("65a92aaa74eaae3120b6f2ac"), "creation_date" : 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.

```
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" } } }
] )
```

```
> 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" } } } ] )
{ "_id" : "D", "sum" : -64 }
{ "_id" : "L", "sum" : 243 }
{ "_id" : "C", "sum" : 176 }
{ "_id" : "E", "sum" : 120 }
{ "_id" : "K", "sum" : 153 }
{ "_id" : "J", "sum" : 419 }
{ "_id" : "O", "sum" : 71 }
{ "_id" : "P", "sum" : 94 }
{ "_id" : "V", "sum" : -43 }
{ "_id" : "A", "sum" : -28 }
{ "_id" : "R", "sum" : 53 }
{ "_id" : "M", "sum" : 516 }
{ "_id" : "S", "sum" : 296 }
{ "_id" : "Z", "sum" : -82 }
{ "_id" : "T", "sum" : -68 }
{ "_id" : "G", "sum" : 199 }
{ "_id" : "H", "sum" : 79 }
{ "_id" : "B", "sum" : 323 }
```

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.

```
set index
expire index 120
```

ttl index
persist index
ttl index

```
127.0.0.1:6379> set index value
OK
127.0.0.1:6379> expire index 120
(integer) 1
127.0.0.1:6379> ttl index
(integer) 104
127.0.0.1:6379> persist index
(integer) 1
127.0.0.1:6379> ttl index
(integer) -1
127.0.0.1:6379>
```

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.

```
zadd rating 10 mysql 20 postgresql 30 mongodb 40 redis
zincrby rating 15 mysql
zpopmax rating
zrem rating redis
zrank rating mysql
```

```
(integer) 1
127.0.0.1:6379> zadd rating 10 mysql 20 postgresql 30 mongodb 40 redis
(integer) 4
127.0.0.1:6379> zincrby rating 15 mysql
"25"
127.0.0.1:6379> zpopmax rating
1) "redis"
2) "40"
127.0.0.1:6379> zrem rating redis
(integer) 0
127.0.0.1:6379> zrank rating mysql
(integer) 1
127.0.0.1:6379>
```

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”.

subscribe events42
publish events42 Hello:there

```
127.0.0.1:6379> subscribe events42
1) "subscribe"
2) "events42"
3) (integer) 1
1) "message"
2) "events42"
3) "Hello"
1) "message"
2) "events42"
3) "Hello there"
127.0.0.1:6379> publish events42 "Hello there"
(integer) 1
```