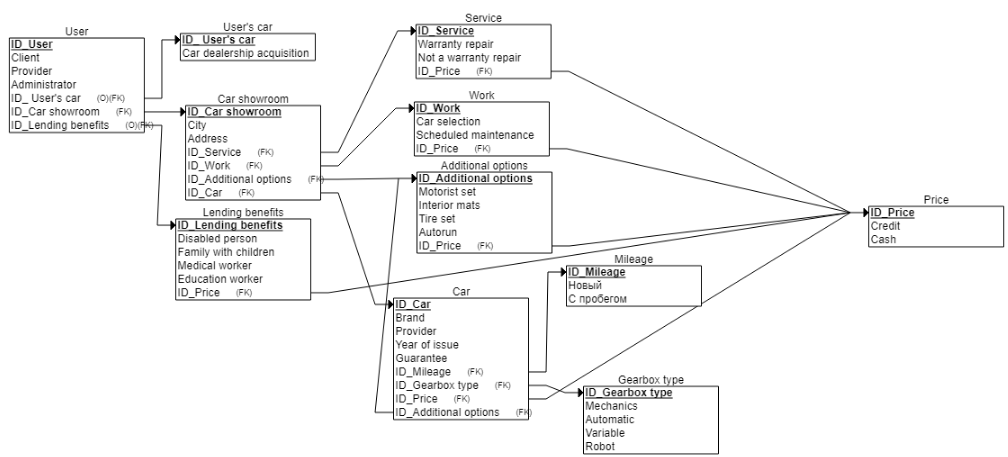
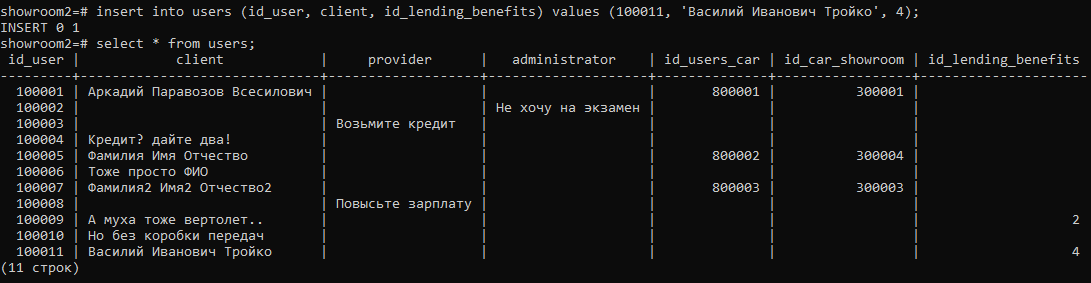
|  |  |  |
| --- | --- | --- |
| Лабораторная работа №4  Реализация SQL-запросов базы данных | Ф.И.О. | Сидский Н. А. |
| Группа | ИВТ-261 |
| Преподаватель | Аль-Мерри Гаис М. С. |
| Дата сдачи |  |



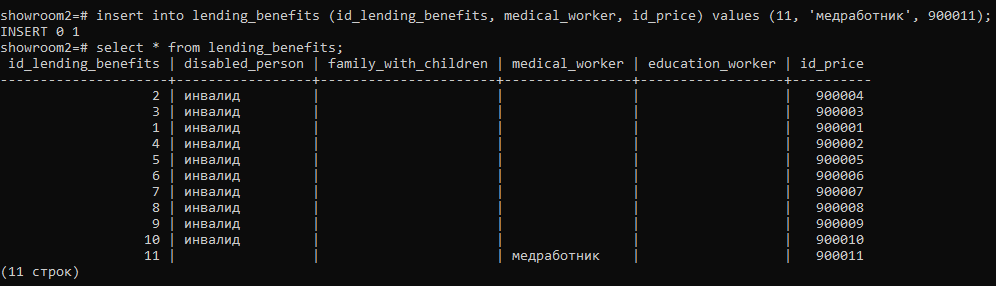
Запросы из функциональных требований:

1. Добавить клиента

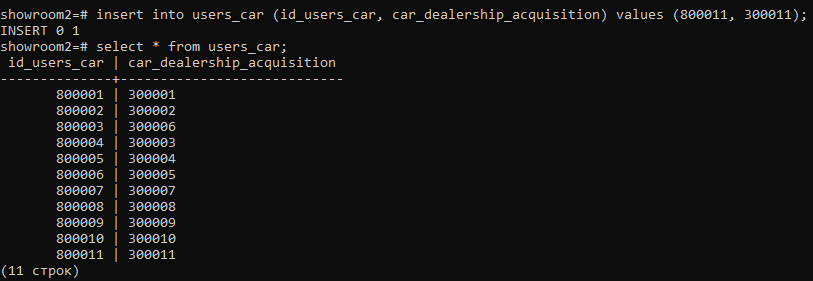
insert into users (id\_user, client, id\_lending\_benefits) values (100011, 'Василий Иванович Тройко', 4);



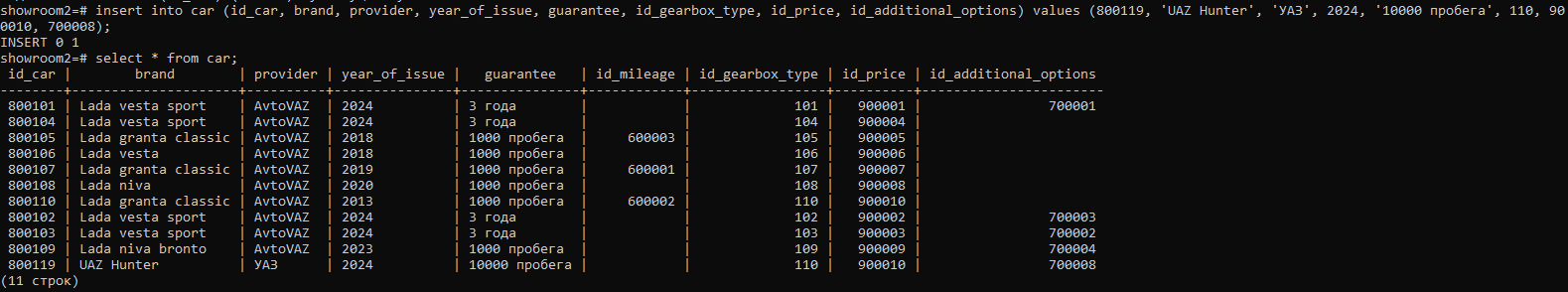
1. Добавить льготу

insert into lending\_benefits (id\_lending\_benefits, medical\_worker, id\_price) values (11, 'медработник', 900011);

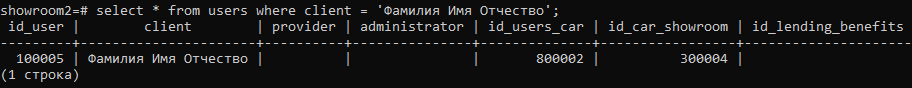
1. Добавить автомобиль пользователя

insert into users\_car (id\_users\_car, car\_dealership\_acquisition) values (800011, 300011); insert into car (id\_car, brand, provider, year\_of\_issue, guarantee, id\_gearbox\_type, id\_price, id\_additional\_options) values (800119, 'UAZ Hunter', 'УАЗ', 2024, '10000 пробега', 110, 900010, 700008);

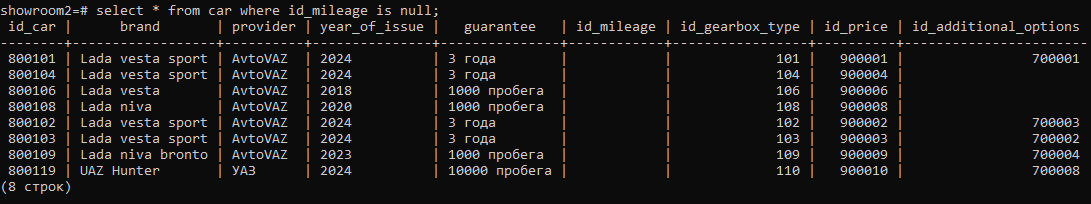
1. Добавить автомобиль в автосалон



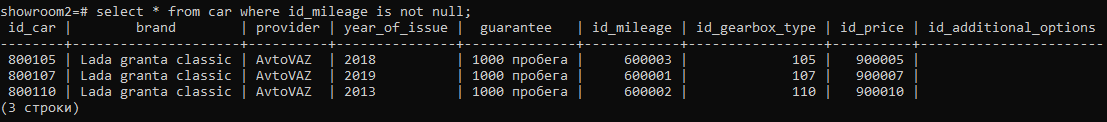
1. Найти пользователя

select \* from users where client = 'Фамилия Имя Отчество';

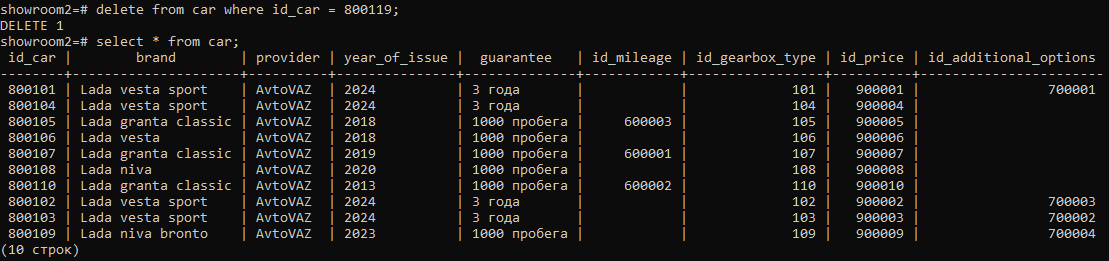
1. Показать новые автомобили в салоне

select \* from car where id\_mileage is null; 

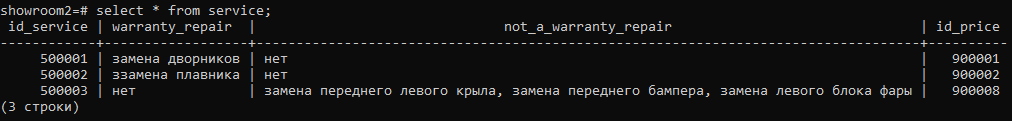
1. Показать автомобили с пробегом в салоне

select \* from car where id\_mileage is not null; 

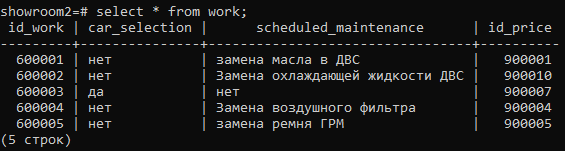
1. Удалить автомобиль

delete from car where id\_car = 800119; 

1. Показать обращения в сервис

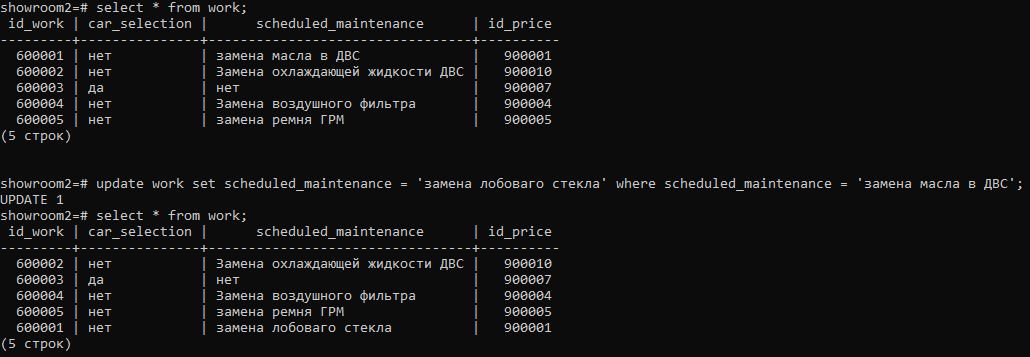
select \* from service;

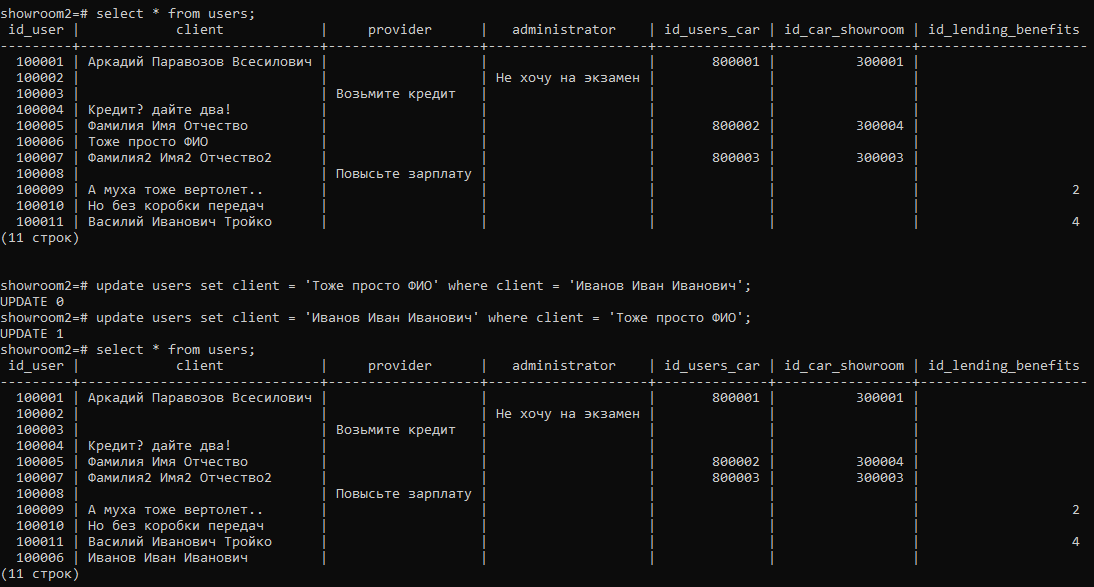
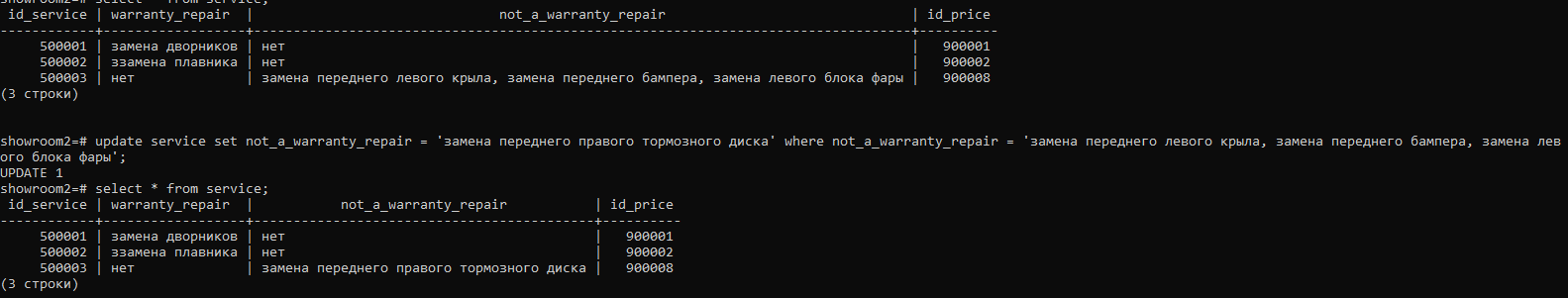
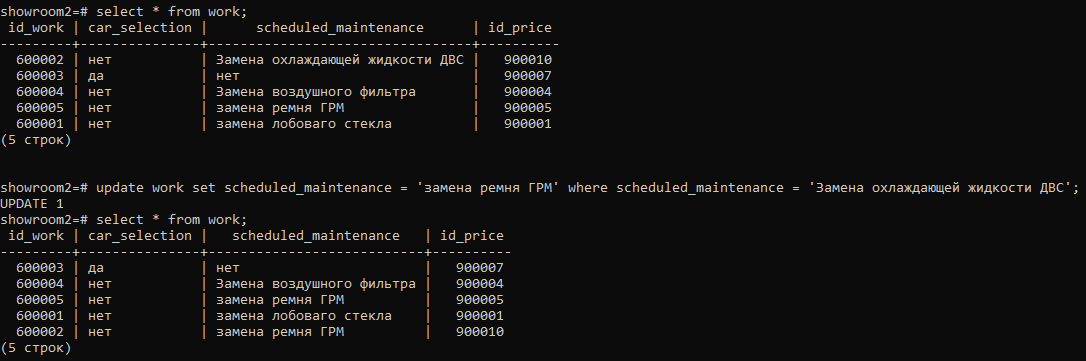
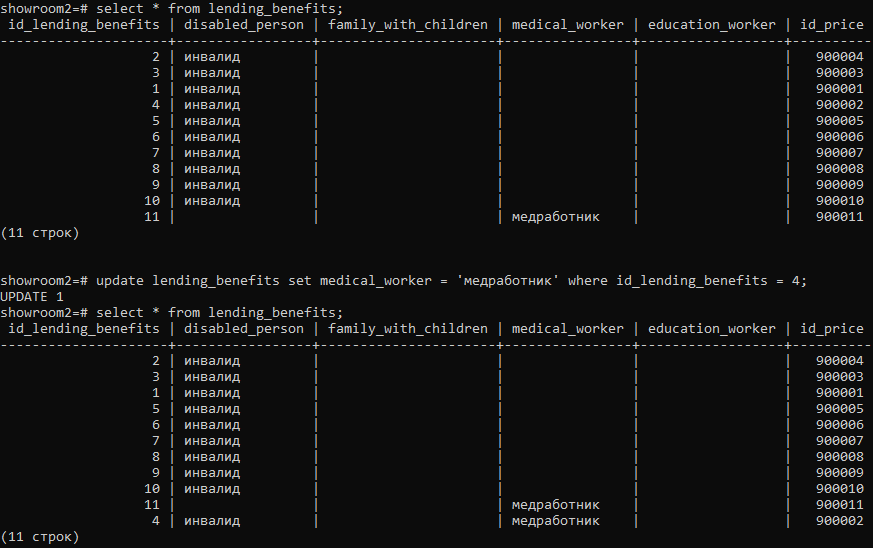
1. Показать проведенные работы

select \* from work;

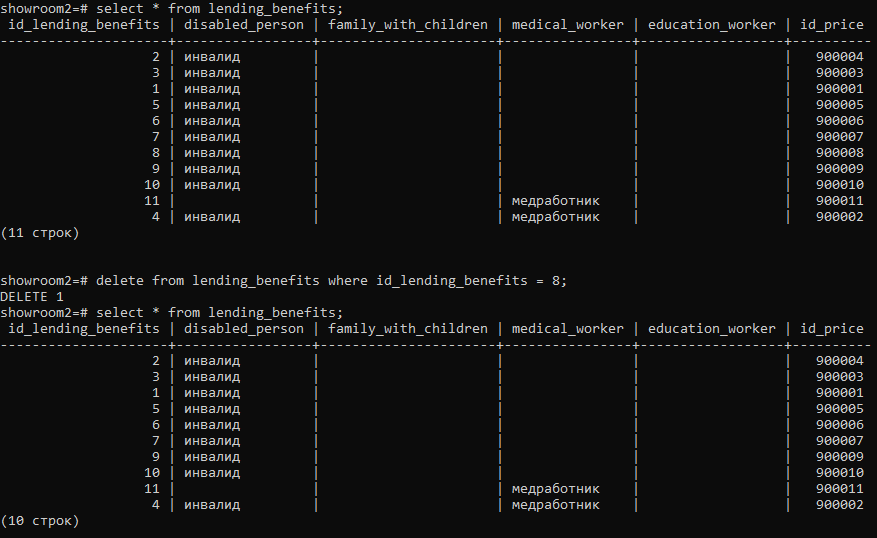
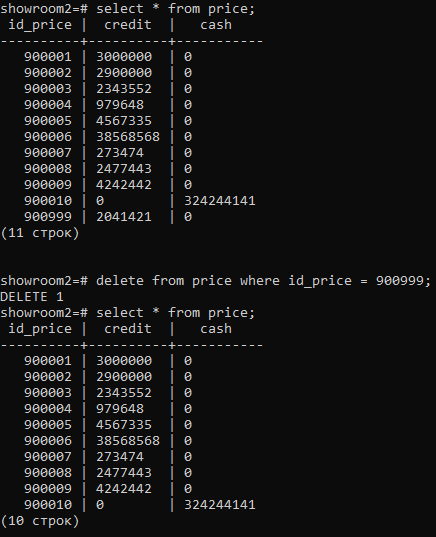
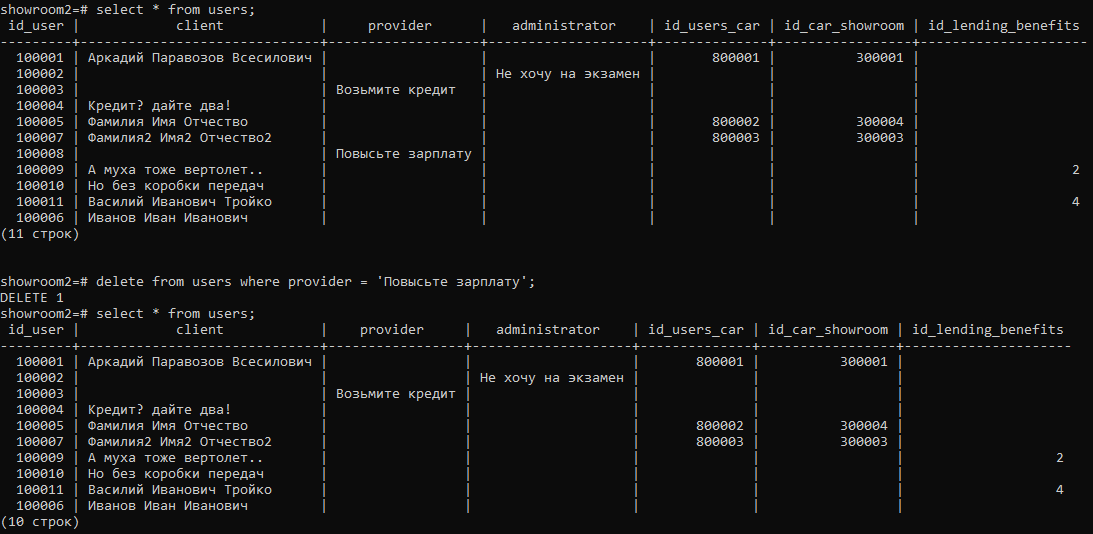
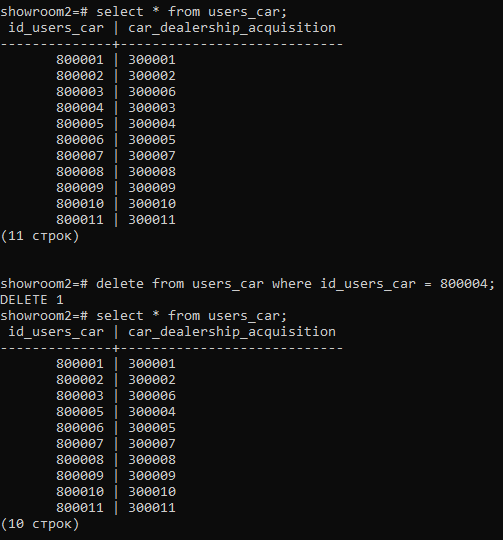
UPDATE и WHERE

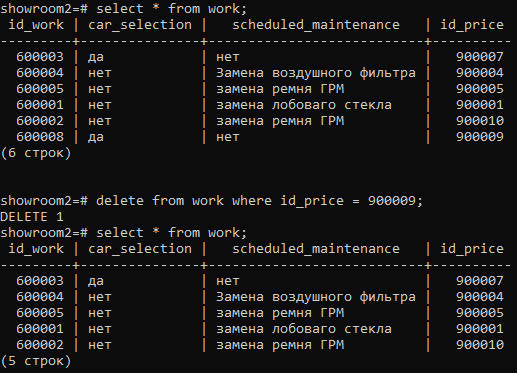
Используется для изменения данных в таблице



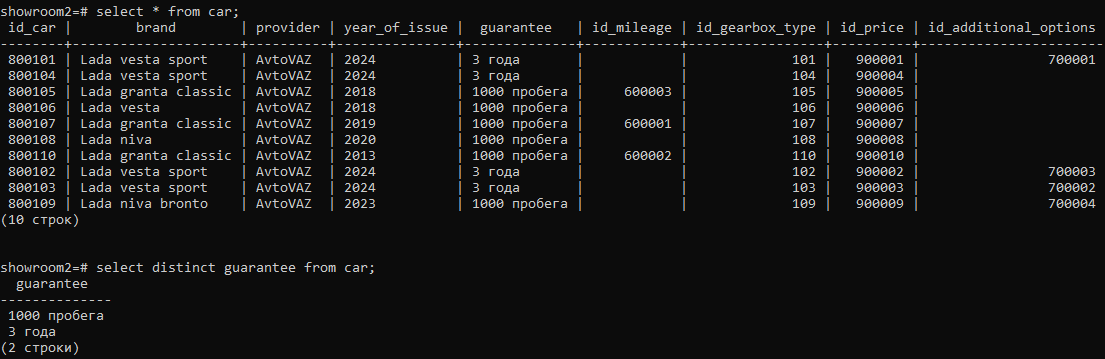
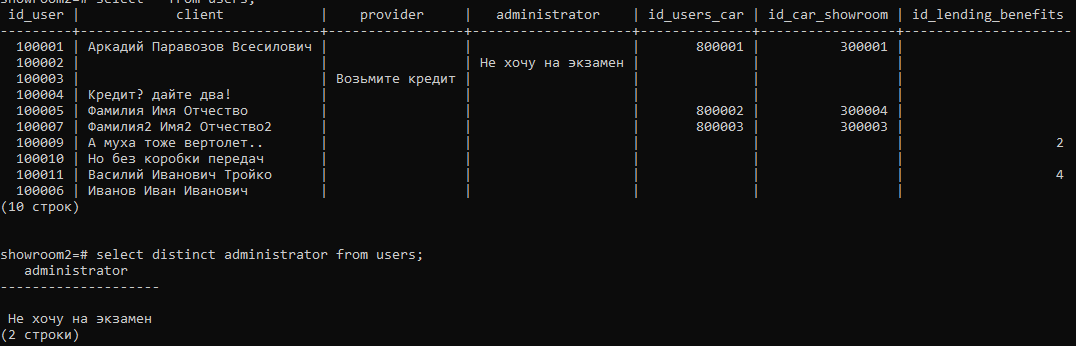
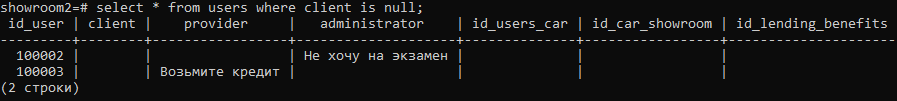
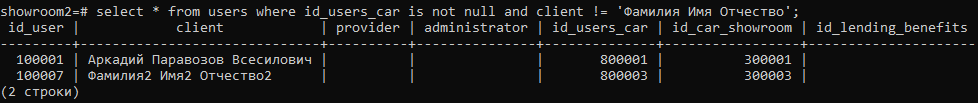
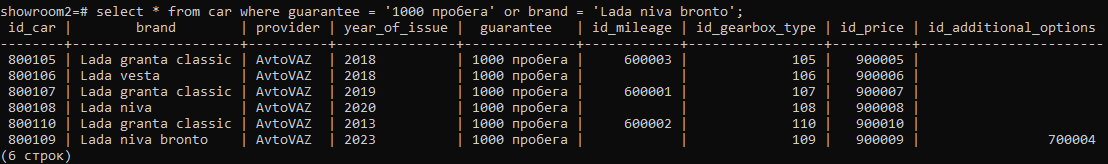
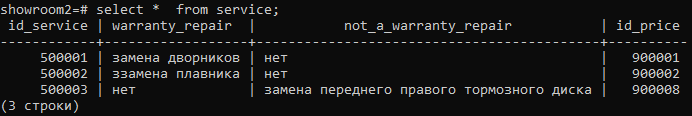
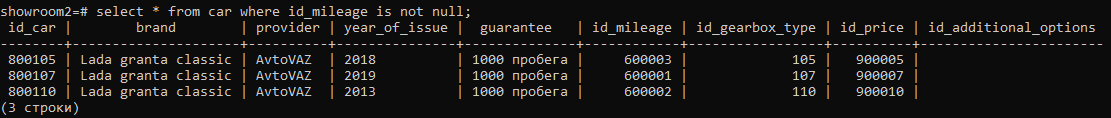
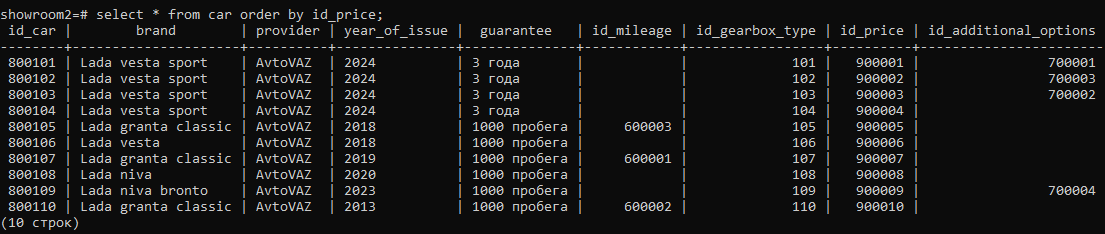
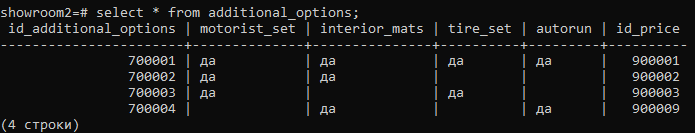
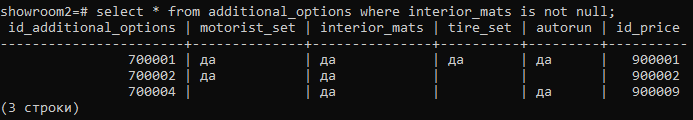
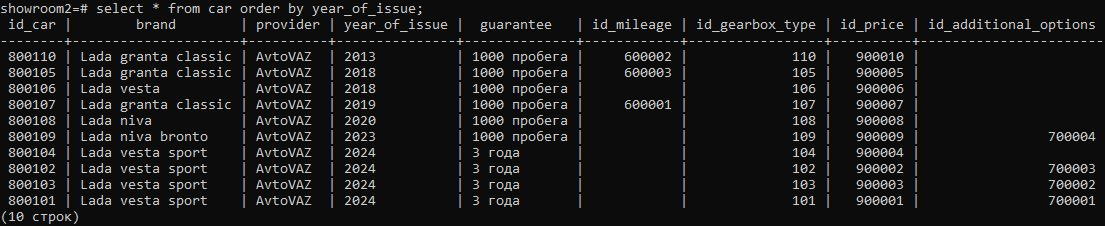
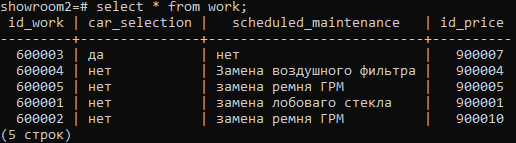
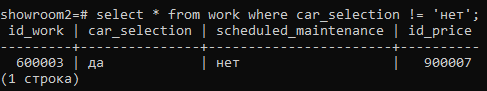
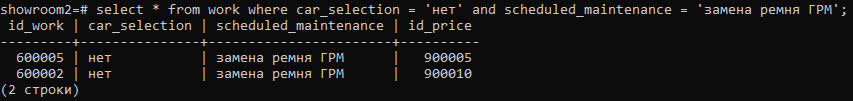
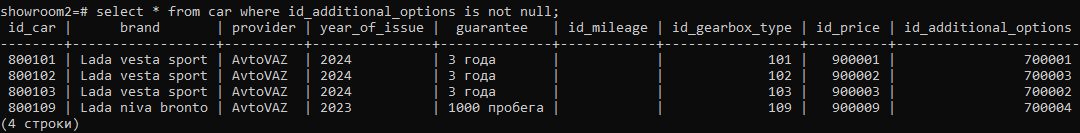
DELETE

Используются для удаления таблиц или их строк/столбов  



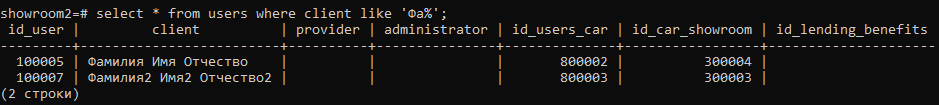
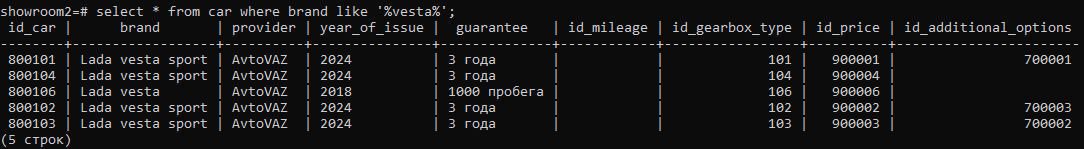
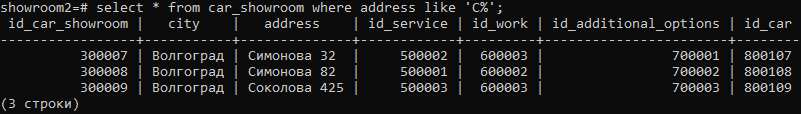
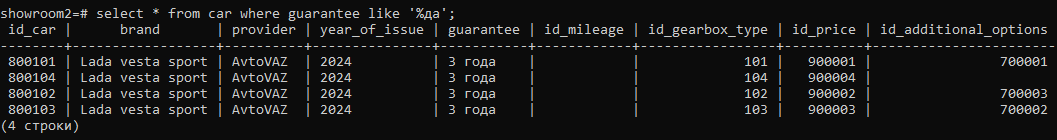
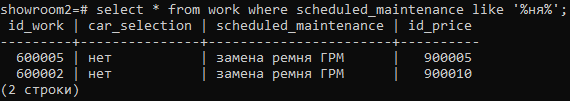
Запросы SELECT, DISTINCT, WHERE, AND/OR/NOT, IN, BETWEEN:

Используются для вывода каких-либо данных из таблиц на экран.

1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 
12. 
13. 
14. 
15. 

LIKE

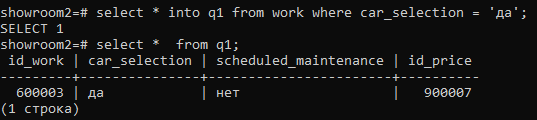
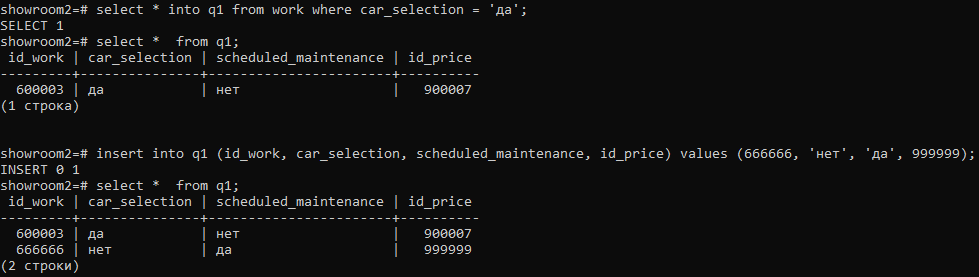
Используется для нахождение определенных строк в таблице по неполным данным или для сравнения значений с выводом true/false

1. 
2. 
3. 
4. 
5. 

SELECT INTO и INSERT INTO

SELECT INTO используется для перенесения значений из уже созданной таблицы в новую.

INSERT INTO используется для занесения данных в уже созданную таблицу.

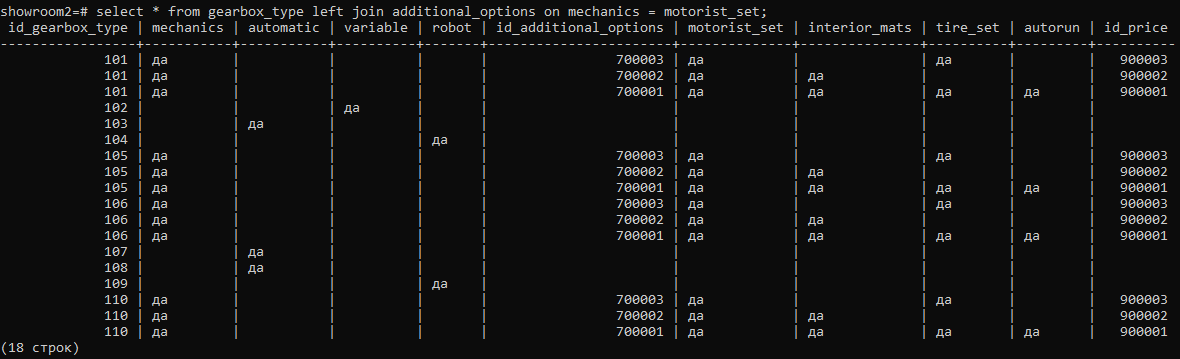
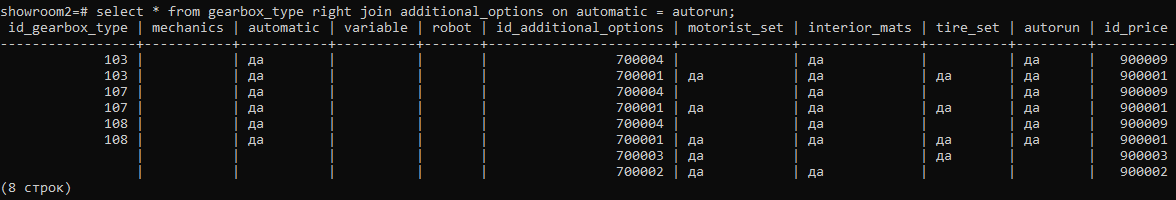
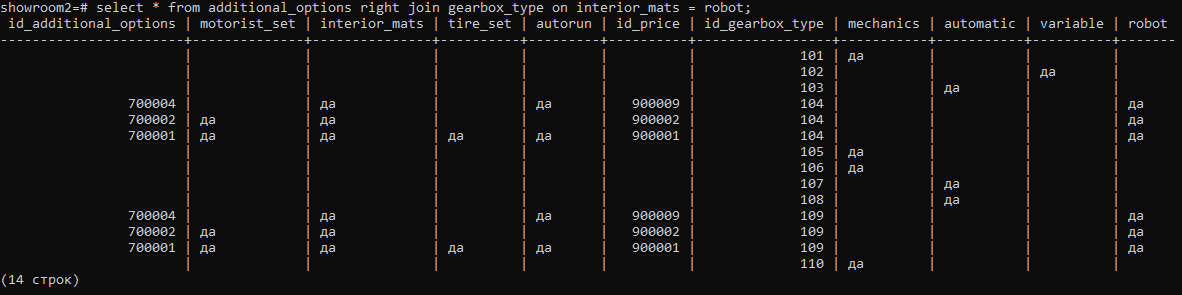
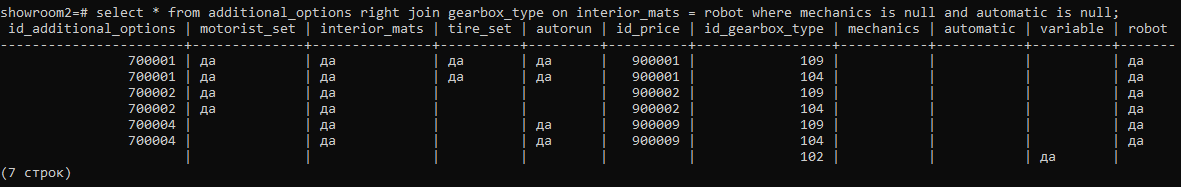
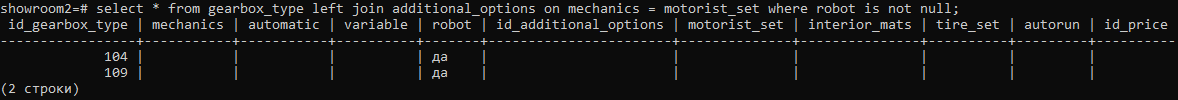
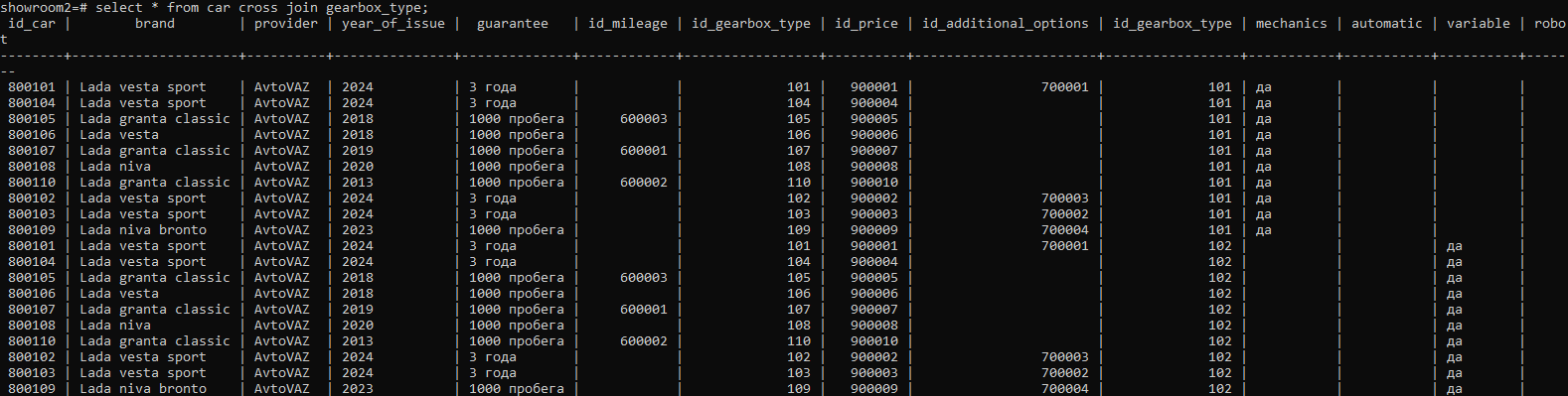
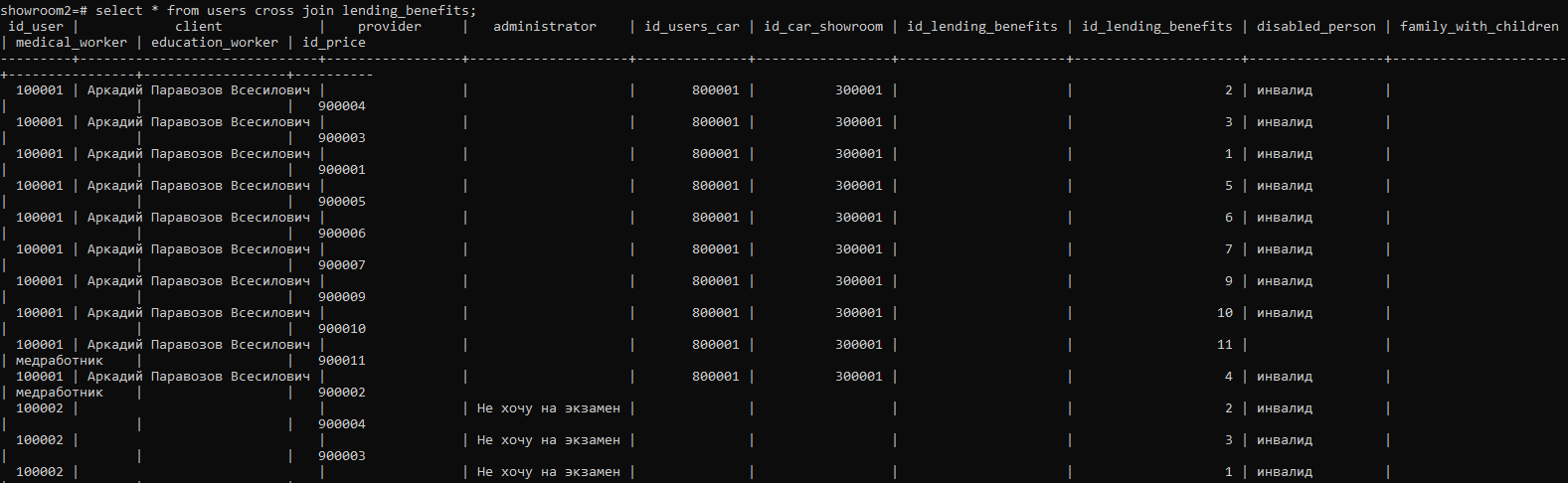
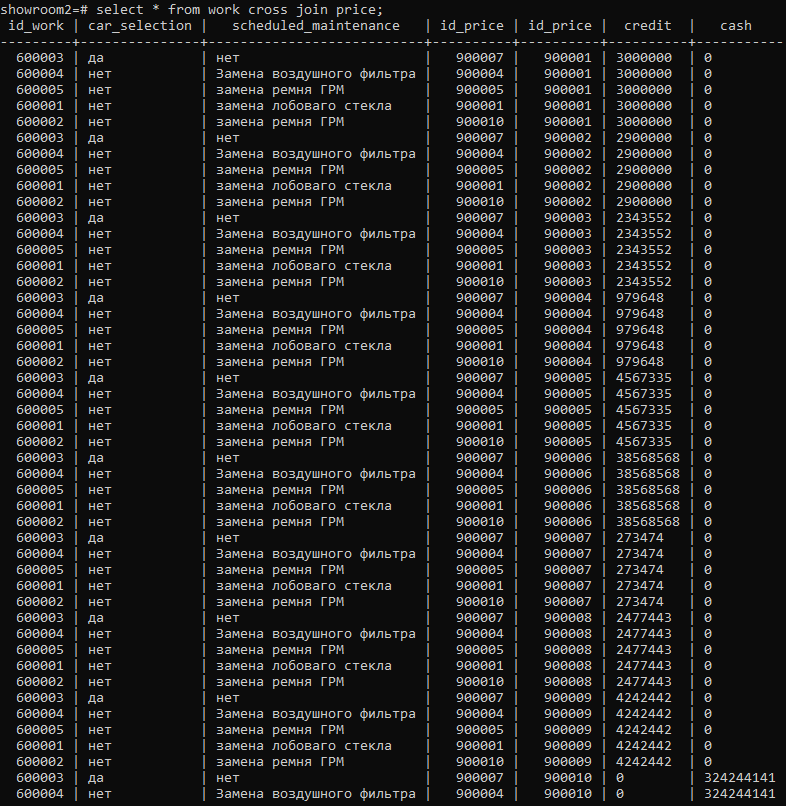
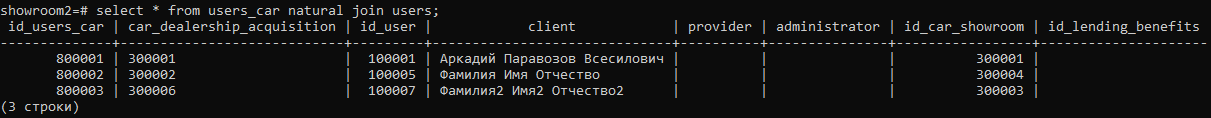
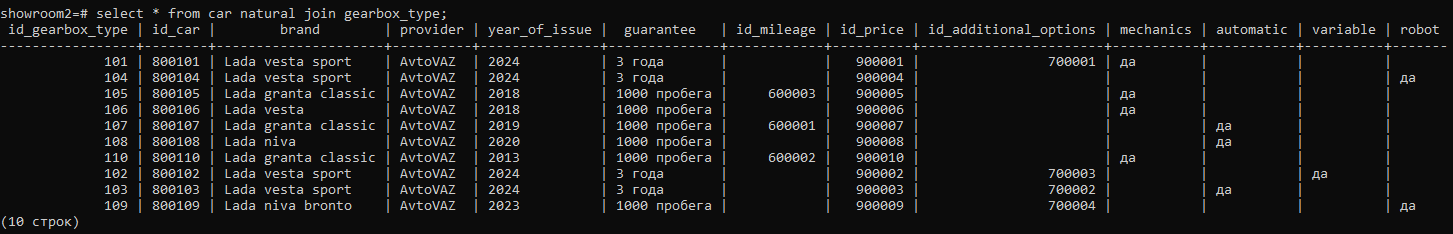
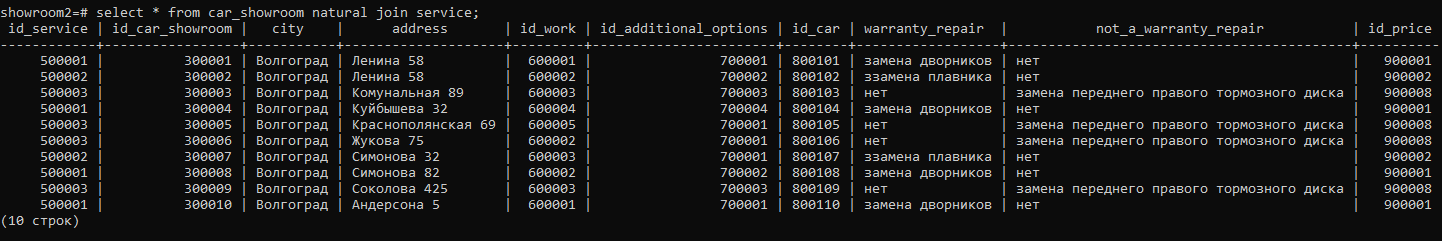
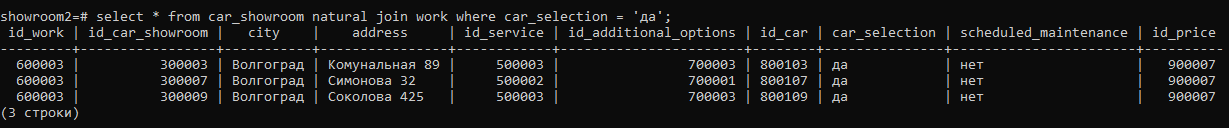
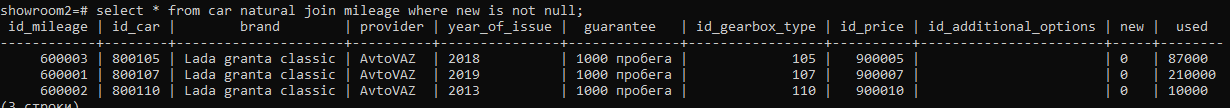
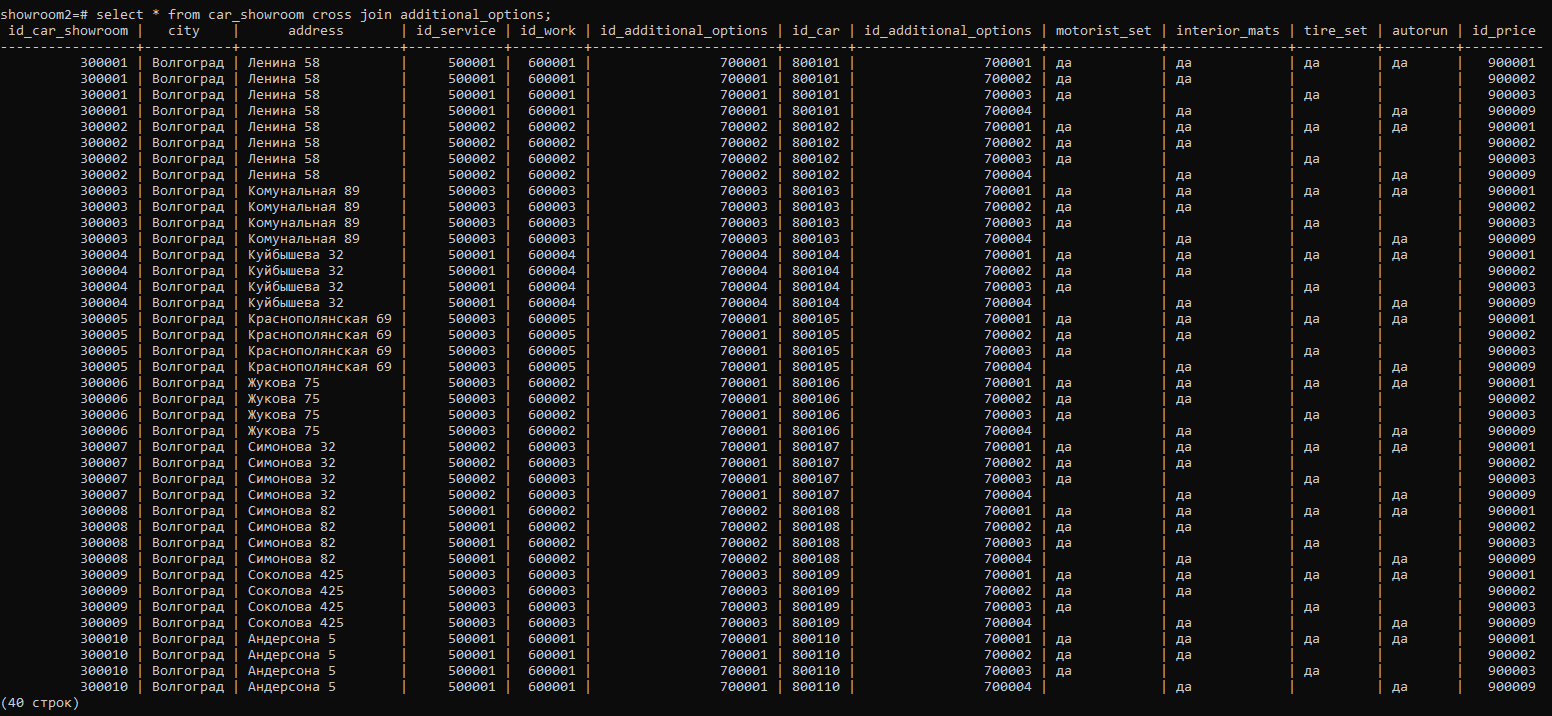
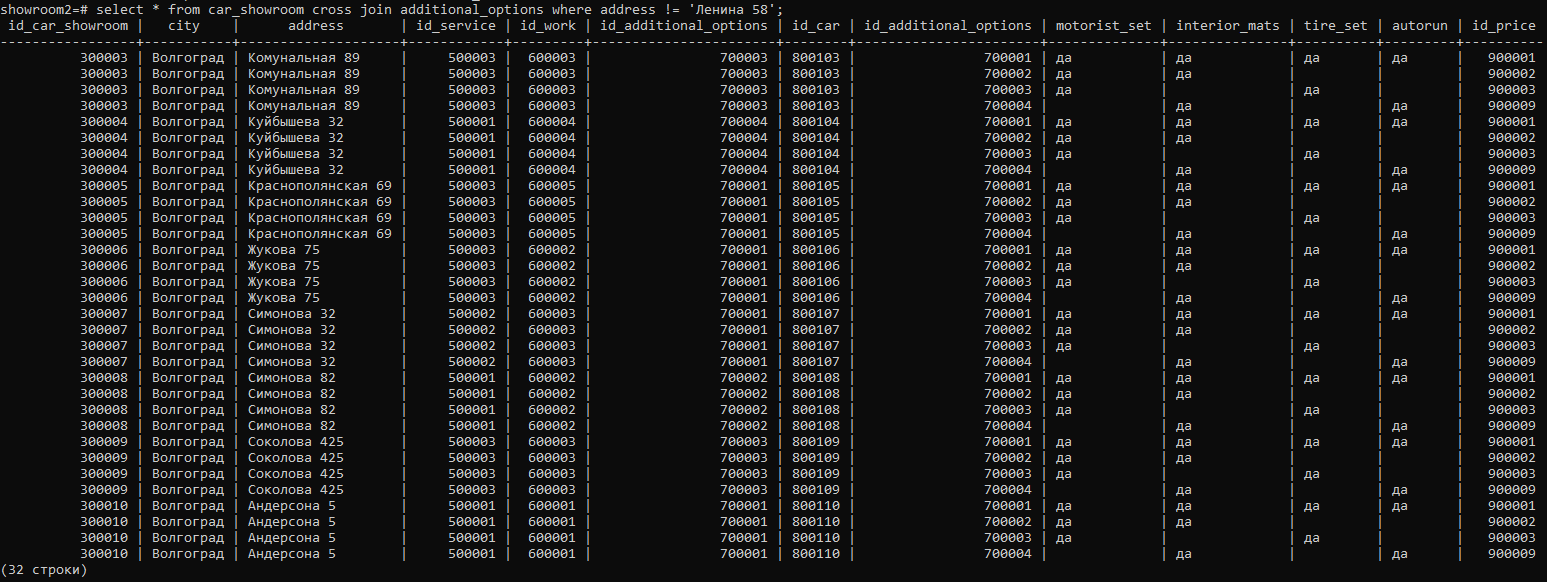
1. 
2. 

JOIN

INNER JOIN используется для объединения строк из двух таблиц, которые имеют общие значения в указанных столбцах.

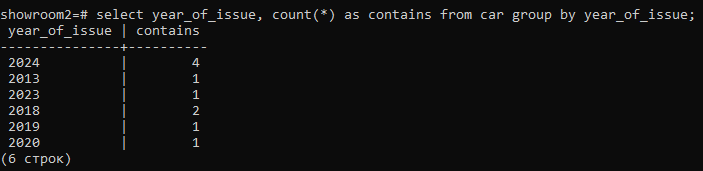
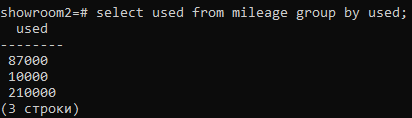
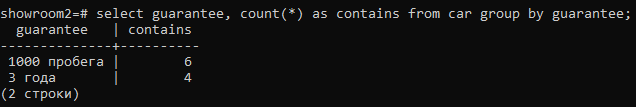
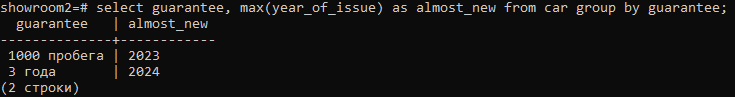
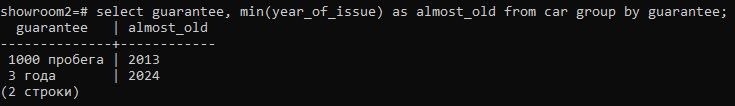
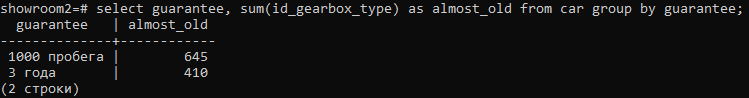
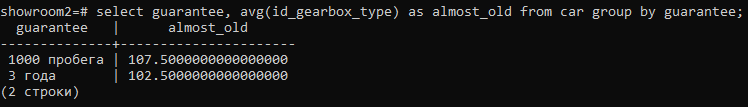
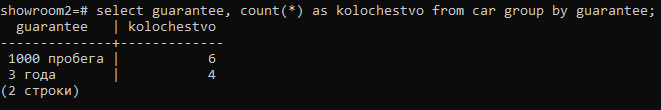
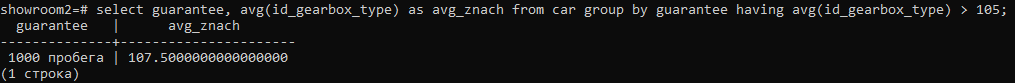
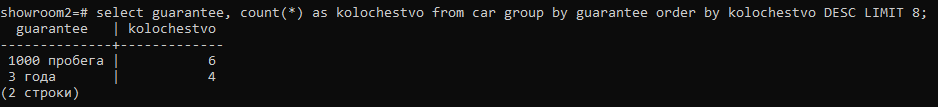
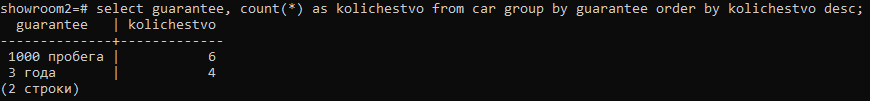
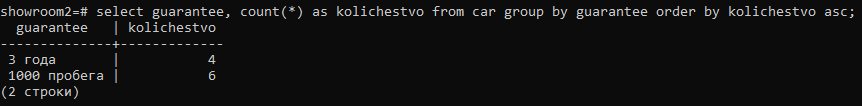
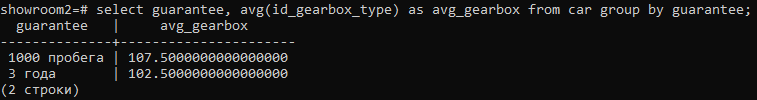
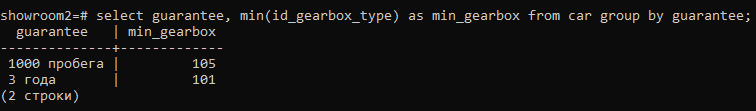
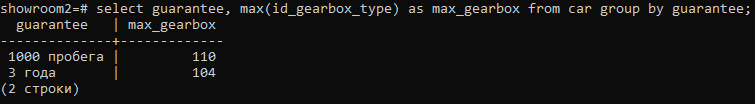
CROSS JOIN используется для получения комбинации всех строк из двух таблиц.

NATURAL JOIN используется для объединения таблиц по всем столбцам с одинаковыми именами.

1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 
12. 
13. 
14. 
15. 

GROUP BY

Используется для объединения строк по категориям или каким-либо параметрам из таблицы.

1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 
12. 
13. 
14. 
15. 

UNION, EXCEPT и INTERSECT

Операторы UNION, EXCEPT и INTERSECT используются для выполнения операций над множествами данных в SQL.

UNION:

UNION используется для объединения результатов двух или более SELECT запросов в один результат.

Оператор UNION удаляет дубликаты из объединенного результата.

Например, UNION может использоваться для объединения результатов запросов, возвращающих данные из разных таблиц или условий.

EXCEPT:

EXCEPT используется для получения разности между результатами двух SELECT запросов.

Оператор EXCEPT возвращает только уникальные строки из первого запроса, которые не присутствуют во втором запросе.

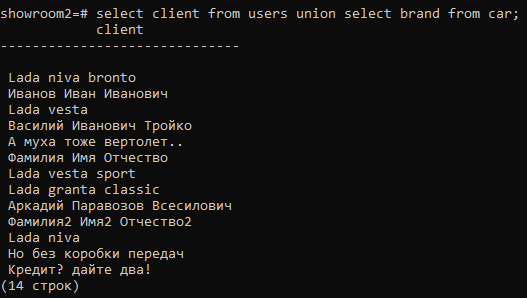
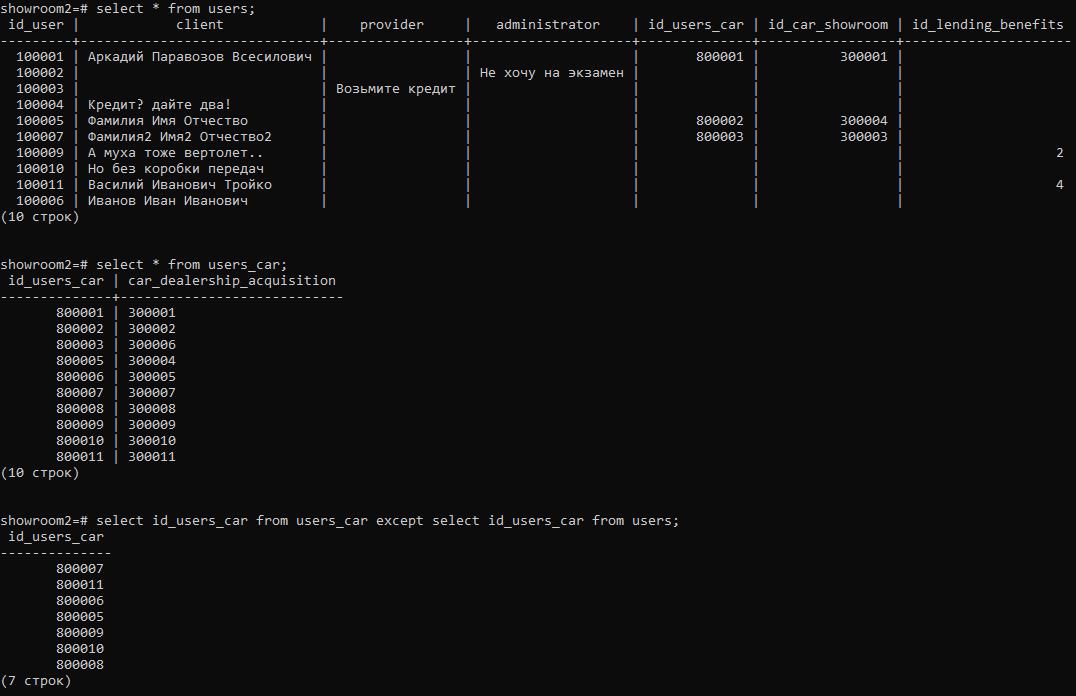
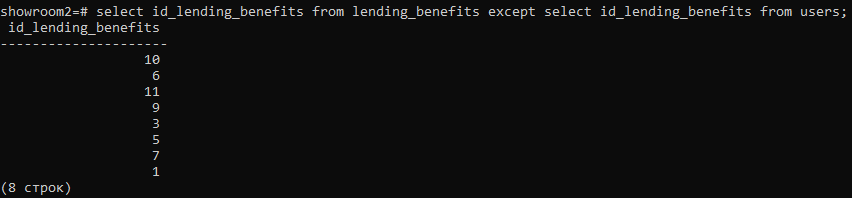
Например, EXCEPT может использоваться для получения всех клиентов, которые не сделали покупок в последний месяц.

INTERSECT:

INTERSECT используется для получения пересечения результатов двух SELECT запросов.

Оператор INTERSECT возвращает только уникальные строки, которые присутствуют и в первом, и во втором запросе.

Например, INTERSECT может использоваться для получения общих элементов из двух списков или таблиц.

1. 
2. 
3. 

Вложенные SELECT с GROUP BY, ALL, ANY, EXISTS

GROUP BY:

GROUP BY используется для группировки строк с одинаковыми значениями в одну строку-сводку.

Он часто используется с агрегатными функциями, такими как COUNT, MAX, MIN, SUM, AVG, чтобы получить сводные данные по группам.

Например, GROUP BY может использоваться для подсчета количества клиентов в каждой стране или для получения суммарных продаж по категориям товаров.

ALL:

ALL используется для сравнения значения с каждым значением в подзапросе.

Оператор ALL возвращает TRUE, если все значения в подзапросе удовлетворяют условию.

Например, ALL может использоваться для проверки, является ли значение больше всех значений в подзапросе.

ANY:

ANY используется для сравнения значения с любым значением в подзапросе.

Оператор ANY возвращает TRUE, если хотя бы одно значение в подзапросе удовлетворяет условию.

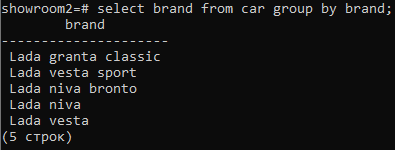
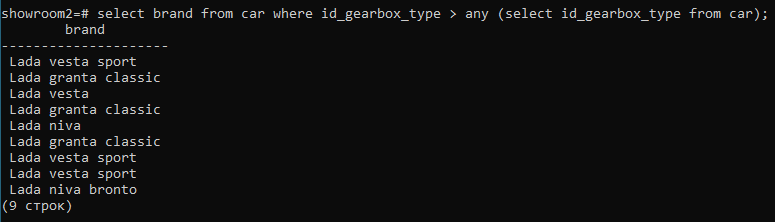
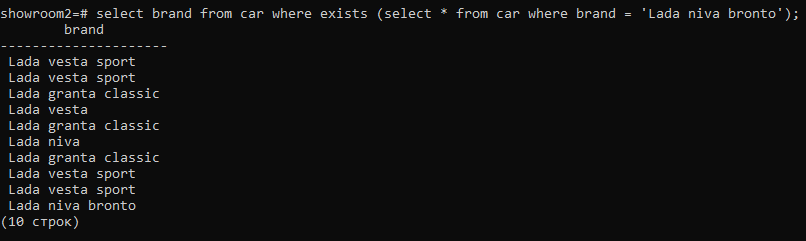
Например, ANY может использоваться для проверки, является ли значение больше любого значения в подзапросе.

EXISTS:

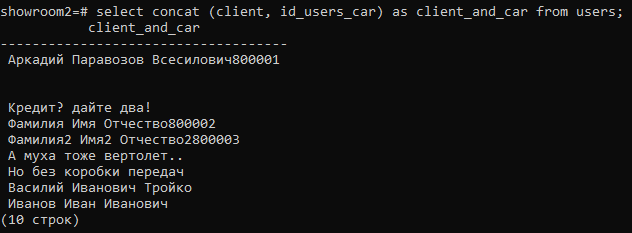
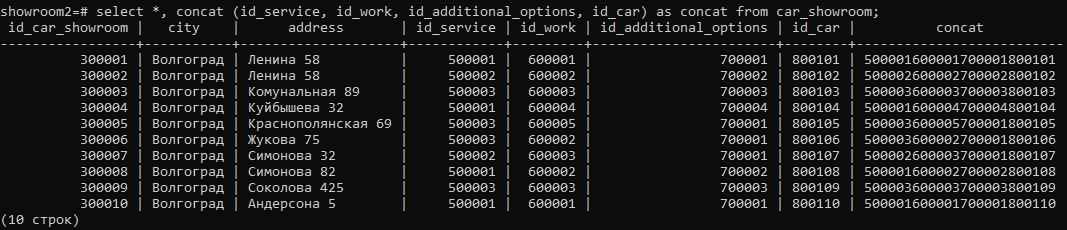
EXISTS используется для проверки существования хотя бы одной строки в подзапросе.

Оператор EXISTS возвращает TRUE, если подзапрос возвращает хотя бы одну строку, и FALSE в противном случае.

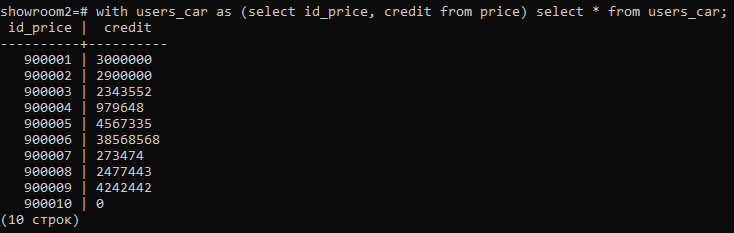
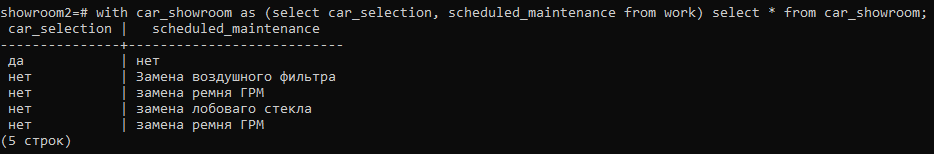
Например, EXISTS может использоваться для проверки, существуют ли заказы для определенного клиента.

1. 
2. 
3. 

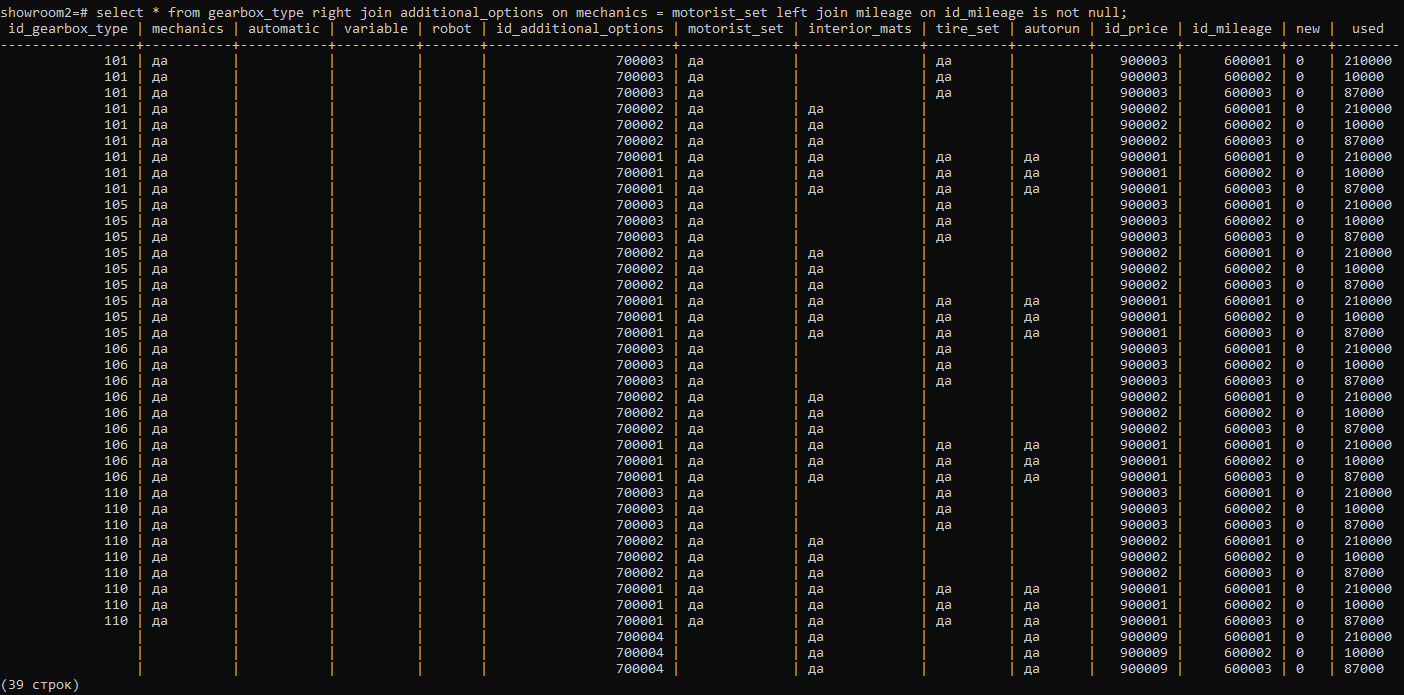
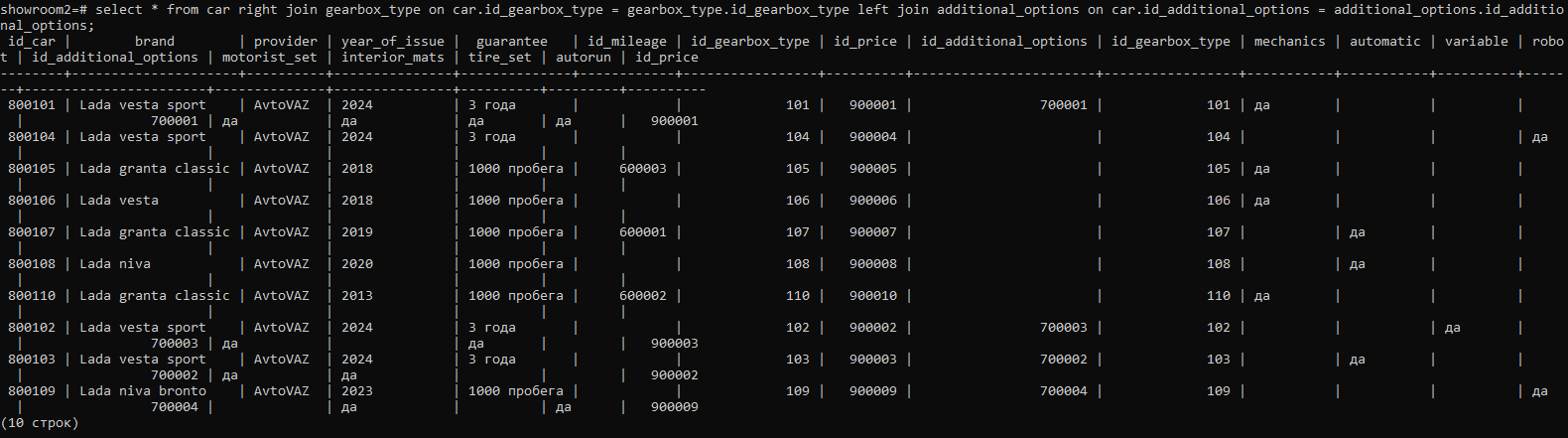
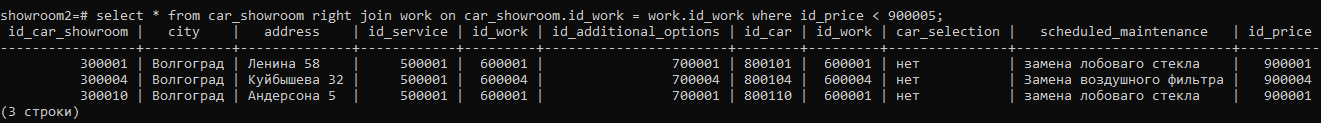
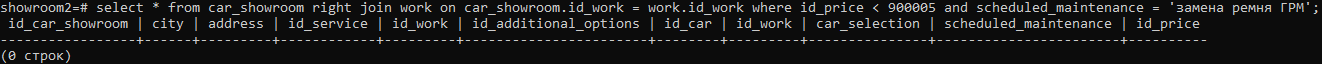
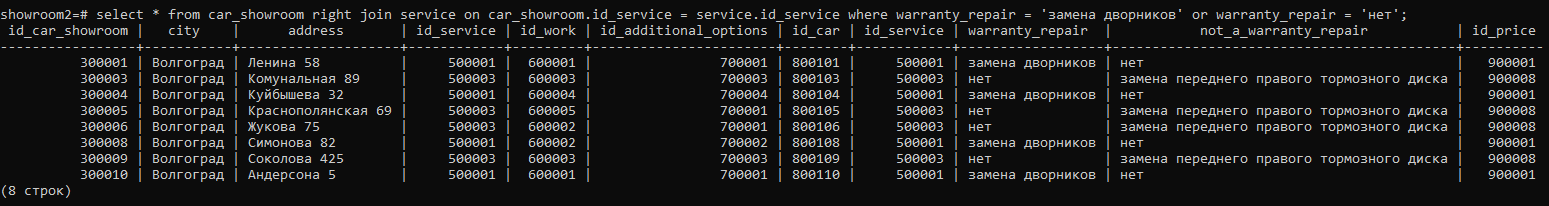
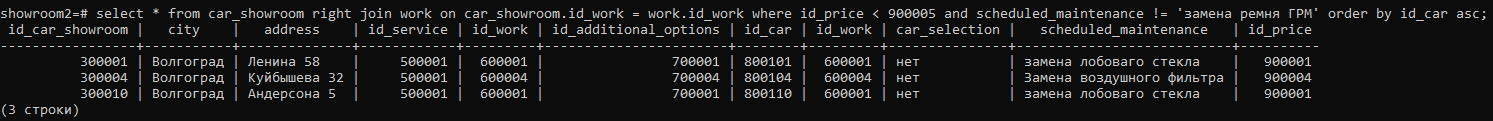
GROUP\_CONCAT

1. 
2. 

Запросы с WITH

1. 
2. 

Сложные запросы:

1. select \* from gearbox\_type right join additional\_options on mechanics = motorist\_set left join mileage on id\_mileage is not null;
2. select \* from car right join gearbox\_type on car.id\_gearbox\_type = gearbox\_type.id\_gearbox\_type left join additional\_options on car.id\_additional\_options = additional\_options.id\_additional\_options;
3. select \* from car\_showroom right join work on car\_showroom.id\_work = work.id\_work where id\_price < 900005;
4. select \* from car\_showroom right join work on car\_showroom.id\_work = work.id\_work where id\_price < 900005 and scheduled\_maintenance = 'замена ремня ГРМ';
5. select \* from car\_showroom right join service on car\_showroom.id\_service = service.id\_service where warranty\_repair = 'замена дворников' or warranty\_repair = 'нет';
6. 
7. 