**Министерство науки и высшего образования Российской Федерации** федеральное государственное автономное образовательное учреждение высшего образования

# «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

**Отчет**

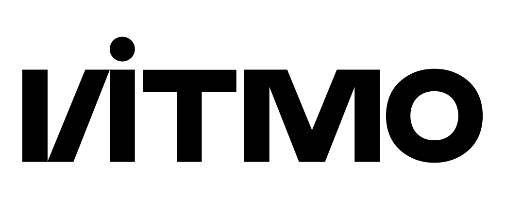
по лабораторной работе №4 «Запросы на выборку и модификацию данных. Представления. Работа с индексами**»**

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Шалунов Андрей Ильич

Факультет: ИКТ

Группа: К3240 Преподаватель: Говорова М.М.



Санкт-Петербург 2023

**Оглавление**

[Цель работы 3](#_Toc149175494)

[Практическое задание 3](#_Toc149175495)

[Вариант 11. БД «Автомастерская» 3](#_Toc149175496)

[Выполнение 3](#_Toc149175497)

[Вывод 6](#_Toc149175498)

## Цель работы

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 1Х, заполнения их рабочими данными, резервного копирования и восстановления БД.

## Практическое задание

Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию).

1. Создать схему в составе базы данных.
2. Создать таблицы базы данных.
3. Установить ограничения на данные: *Primary Key, Unique, Check, Foreign Key*.
4. Заполнить таблицы БД рабочими данными.
5. Создать резервную копию БД. *Указание:*

*Создать две резервные копии:*

* *с расширением CUSTOM для восстановления БД;*
* *с расширением PLAIN для листинга (в отчете);*
* *при создании резервных копий БД настроить параметры Dump options для Type of objects и Queries .*

7. Восстановить БД.

## Вариант 11. БД «Автомастерская»

**Описание предметной области:**

Описание предметной области: Сеть автомастерских осуществляет ремонт автомобилей, используя для этих целей штат мастеров и свои мастерские. Стоимость ремонта включает цену деталей и стоимость работы.

Заработная плата мастеров составляет 50% стоимости работы.

С клиентом заключается договор на выполнение авторемонтных и профилактических работ, который сопровождается администратором. В каждом договоре может быть несколько видов услуг. Для выполнения видов работ могут требоваться детали или расходные материалы, которые предоставляет либо клиент, либо автомастерская. Если детали предоставляет автомастерская, то их стоимость включается в смету по договору.

Каждый вид работ могут выполнять разные мастера, в зависимости от их специализации. Распределение мастеров выполняет администратор.

БД должна содержать следующий минимальный набор сведений: Табельный номер сотрудника. ФИО сотрудника. Должность. Разряд мастера. Специализация. Адрес автомастерской. Дата заказа. Гос. Номер автомобиля. Марка. Мощность автомобиля. Год выпуска. Цвет автомобиля. Дата принятия в ремонт. Плановая дата окончания ремонта. Фактическая дата окончания ремонта. Вид ремонта. Стоимость вида ремонта. Название детали. Цена детали. Марка и модель автомобиля. Страна производителя. Госномер автомобиля. ФИО владельца. Номер телефона владельца.  E-mail владельца.

## Выполнение

Название создаваемой БД – «Автомастерская» («Autorepair Shop»)

Схема логической модели базы данных, которая сгенерирована в Generate ERD, находится на рисунке 1.

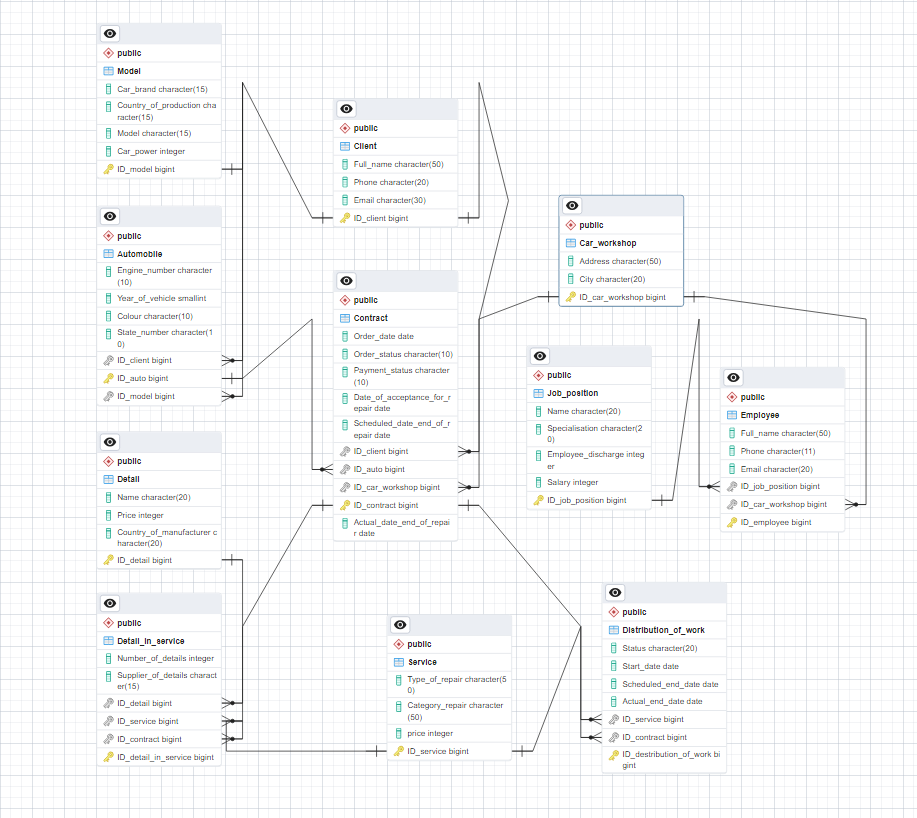


Рисунок 1 — ERD-схема базы данных

**Запросы**

**Выбрать фамилию того механика, который чаще всех работает с автомобилями марки ”Тойота”.**

SELECT e."Full\_name" AS "Механик", jp."Name" AS "Должность", COUNT(c."ID\_contract") AS "Количество работ"

FROM public."Employee" e

JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

JOIN public."Automobile" a ON c."ID\_auto" = a."ID\_auto"

JOIN public."Model" m ON a."ID\_model" = m."ID\_model"

JOIN public."Job\_position" jp ON e."ID\_job\_position" = jp."ID\_job\_position"

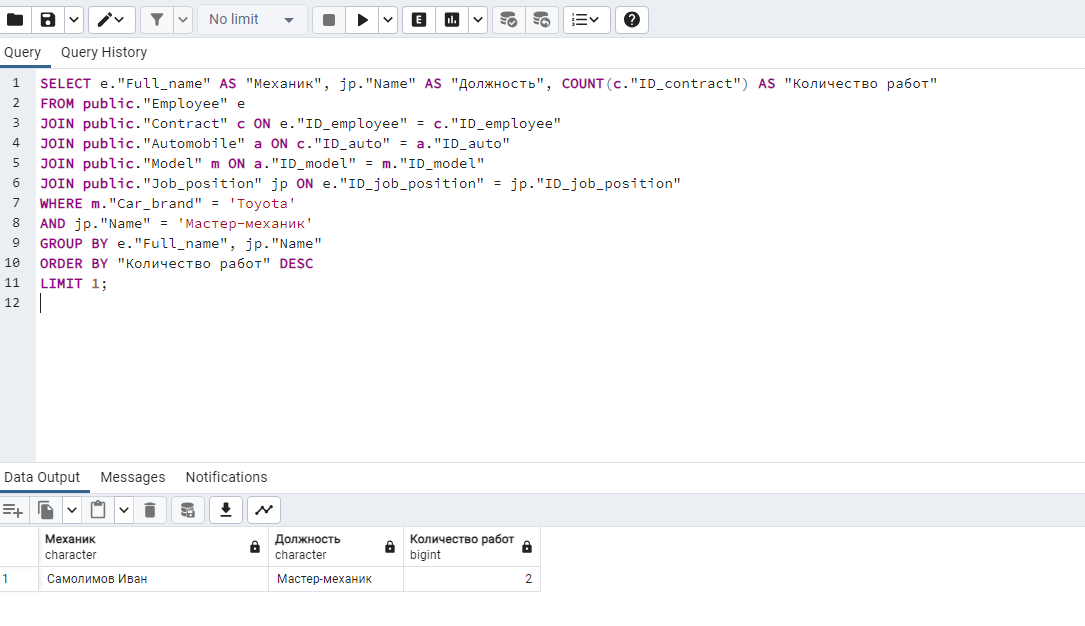
WHERE m."Car\_brand" = 'Toyota'

AND jp."Name" = 'Мастер-механик'

GROUP BY e."Full\_name", jp."Name"

ORDER BY "Количество работ" DESC

LIMIT 1;



**Определить тех владельцев автомобилей, которых всегда обслуживает один и тот же механик. Вывести фамилии механика и его постоянного клиента.**

SELECT

e."Full\_name" AS "Механик",

c."ID\_client" AS "Постоянный клиент"

FROM public."Employee" e

JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

WHERE e."ID\_job\_position" = (SELECT "ID\_job\_position" FROM public."Job\_position" WHERE "Name" = 'Мастер-механик')

AND c."ID\_client" IN (

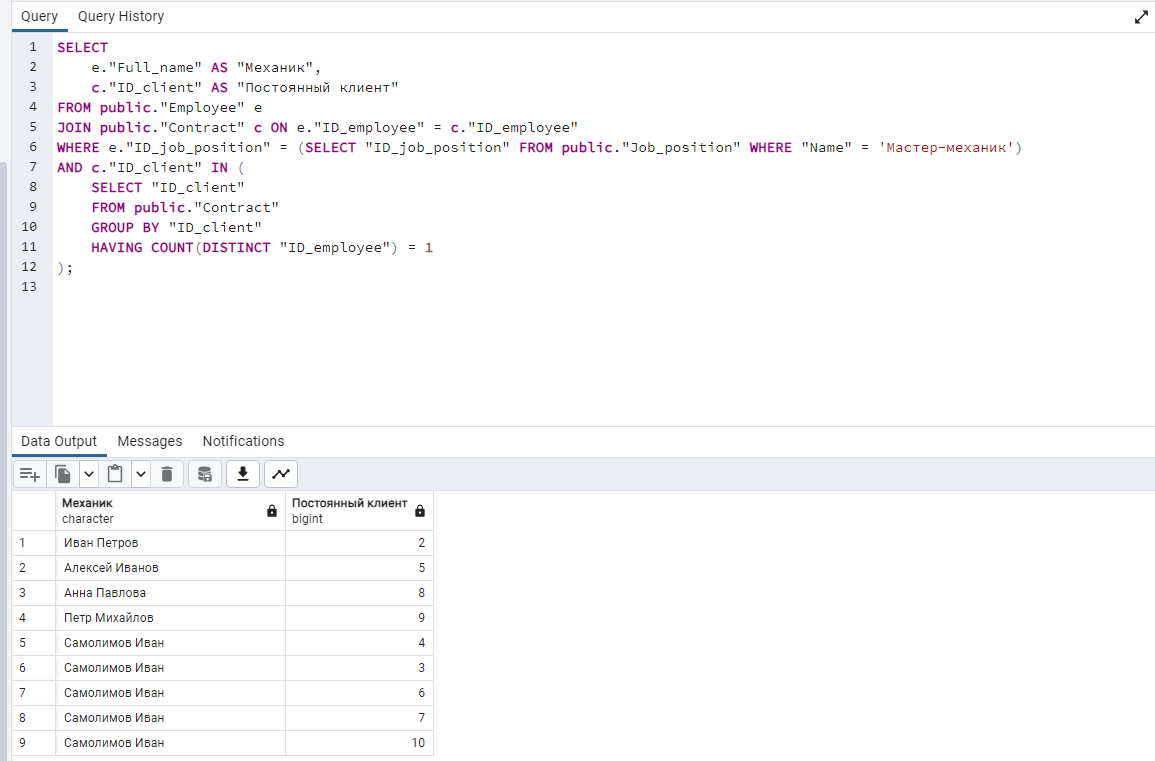
SELECT "ID\_client"

FROM public."Contract"

GROUP BY "ID\_client"

HAVING COUNT(DISTINCT "ID\_employee") = 1

);



**Вывести фамилии механиков, которые не выполняли работы в срок и количество дней просрочки выполнения заказа.**

SELECT

e."Full\_name" AS "Механик",

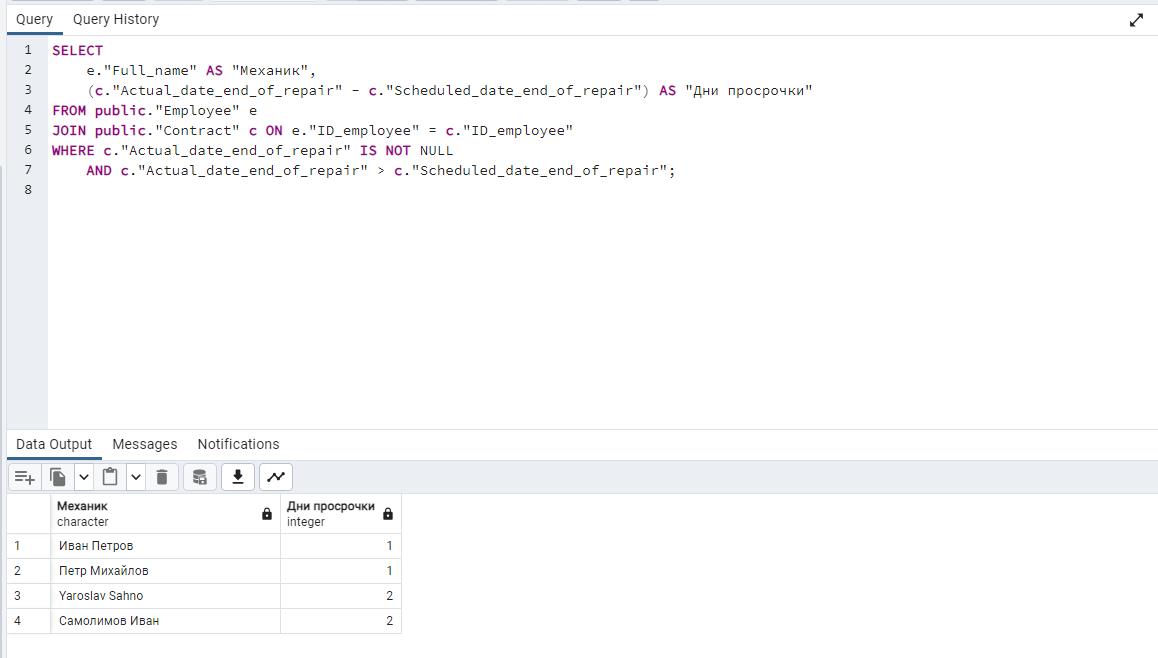
(c."Actual\_date\_end\_of\_repair" - c."Scheduled\_date\_end\_of\_repair") AS "Дни просрочки"

FROM public."Employee" e

JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

WHERE c."Actual\_date\_end\_of\_repair" IS NOT NULL

AND c."Actual\_date\_end\_of\_repair" > c."Scheduled\_date\_end\_of\_repair";



**Вывести данные механика, который выполнял все виды ремонта за прошедшую неделю.**

SELECT e."Full\_name"

FROM public."Employee" e

INNER JOIN public."Job\_position" j ON e."ID\_job\_position" = j."ID\_job\_position"

INNER JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

INNER JOIN public."Distribution\_of\_work" d ON c."ID\_contract" = d."ID\_contract"

INNER JOIN public."Service" s ON d."ID\_service" = s."ID\_service"

WHERE j."Name" = 'Мастер-механик'

AND EXTRACT(WEEK FROM d."Start\_date") = EXTRACT(WEEK FROM CURRENT\_DATE)

AND EXTRACT(YEAR FROM d."Start\_date") = EXTRACT(YEAR FROM CURRENT\_DATE)

GROUP BY e."Full\_name"

HAVING COUNT(DISTINCT s."Type\_of\_repair") = (SELECT COUNT(DISTINCT "Type\_of\_repair") FROM public."Service");



**Сколько заработал каждый мастер за прошедший месяц?**

SELECT e."Full\_name", SUM(c."Total\_payment") as "Total\_earnings"

FROM public."Employee" e

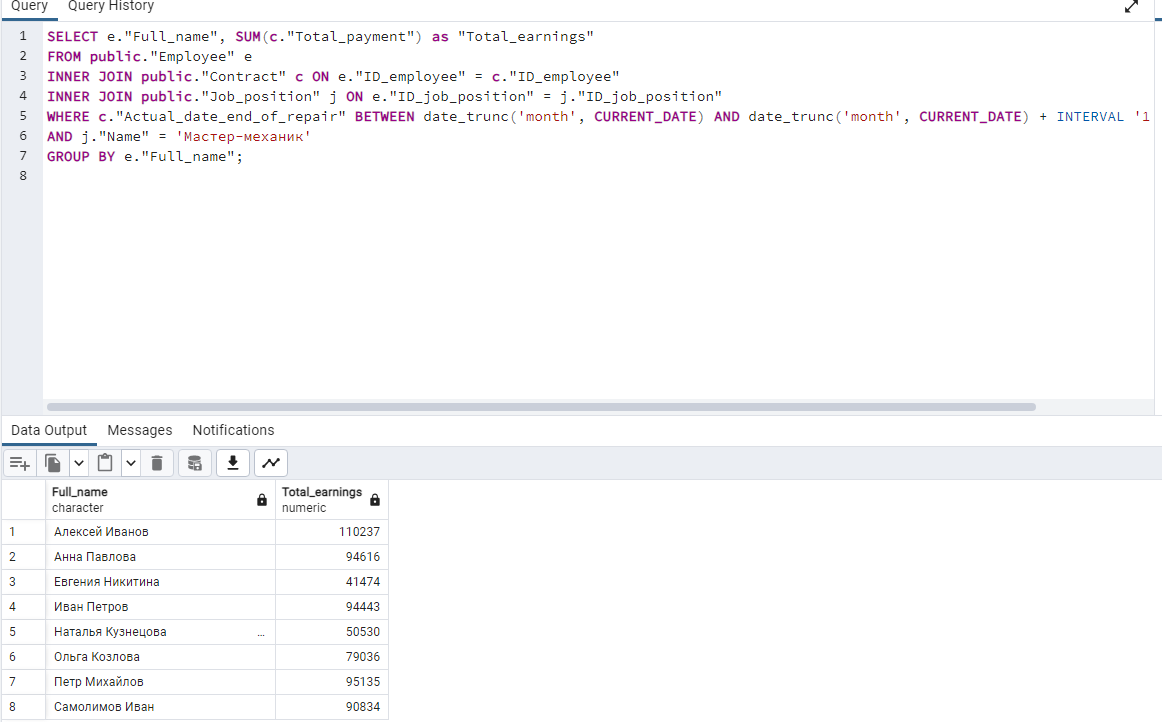
INNER JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

INNER JOIN public."Job\_position" j ON e."ID\_job\_position" = j."ID\_job\_position"

WHERE c."Actual\_date\_end\_of\_repair" BETWEEN date\_trunc('month', CURRENT\_DATE) AND date\_trunc('month', CURRENT\_DATE) + INTERVAL '1 month - 1 day'

AND j."Name" = 'Мастер-механик'

GROUP BY e."Full\_name";



**Вывести данные владельцев автомобилей, которые обращались в ремонт больше одного раза.**

SELECT

c."Full\_name" AS "Владелец",

c."Phone" AS "Телефон",

c."Email" AS "Почта",

a."State\_number" AS "Госномер автомобиля",

m."Car\_brand" AS "Компания",

m."Model" AS "Модель",

COUNT(co."ID\_client") AS "Количество посещений"

FROM public."Client" c

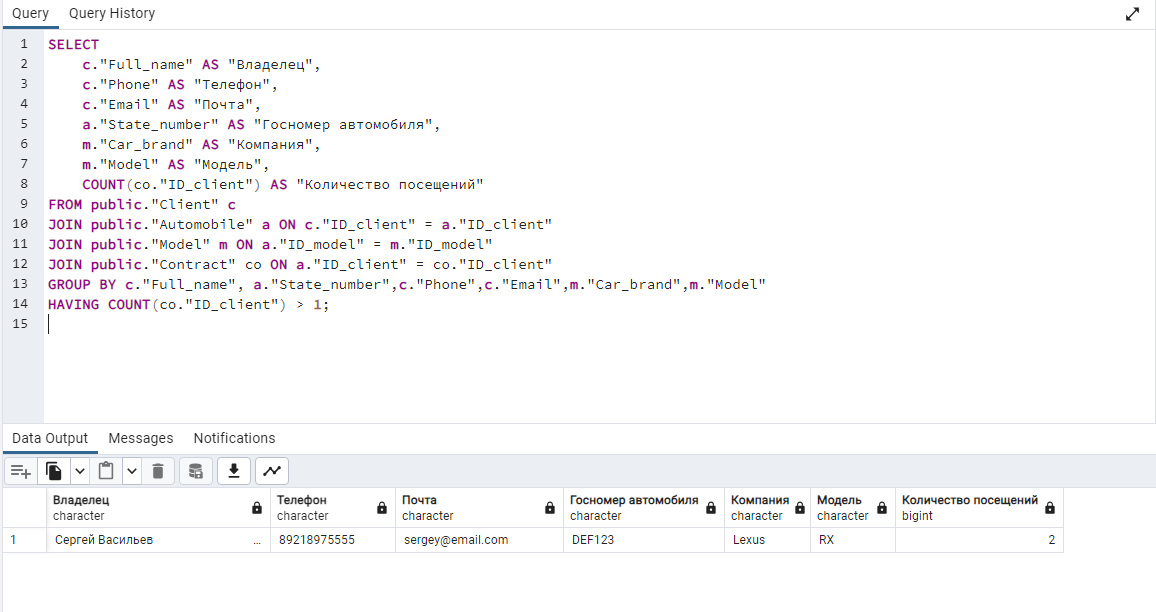
JOIN public."Automobile" a ON c."ID\_client" = a."ID\_client"

JOIN public."Model" m ON a."ID\_model" = m."ID\_model"

JOIN public."Contract" co ON a."ID\_client" = co."ID\_client"

GROUP BY c."Full\_name", a."State\_number",c."Phone",c."Email",m."Car\_brand",m."Model"

HAVING COUNT(co."ID\_client") > 1;



**За каждый день просрочки выполнения заказа механику назначается штраф в размере 5%. Рассчитать штраф каждого механика за прошедший месяц.**

SELECT

e."Full\_name" AS "Механик",

SUM(

CASE

WHEN c."Actual\_date\_end\_of\_repair" > c."Scheduled\_date\_end\_of\_repair"

THEN (c."Actual\_date\_end\_of\_repair" - c."Scheduled\_date\_end\_of\_repair") \* 5

ELSE 0

END

) || '%' AS "Сумма штрафов процентах"

FROM public."Employee" e

JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

JOIN public."Job\_position" jp ON e."ID\_job\_position" = jp."ID\_job\_position"

WHERE c."Actual\_date\_end\_of\_repair" >= current\_date - interval '1 month'

AND jp."Name" = 'Мастер-механик'

GROUP BY e."Full\_name";



**Задание 3.** Создать представление:

Для заказчиков (фамилию механика и модель автомобиля, которую он ремонтирует чаще всего)

CREATE VIEW most\_common\_repair AS

SELECT

sub."Mechanic\_name",

sub."Most\_common\_model"

FROM

(SELECT

e."Full\_name" AS "Mechanic\_name",

m."Model" AS "Most\_common\_model",

COUNT(a."ID\_auto") AS "Number\_of\_repairs",

ROW\_NUMBER() OVER(PARTITION BY e."ID\_employee" ORDER BY COUNT(a."ID\_auto") DESC) as rn

FROM

public."Employee" e

INNER JOIN public."Job\_position" j ON e."ID\_job\_position" = j."ID\_job\_position"

INNER JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

INNER JOIN public."Automobile" a ON c."ID\_auto" = a."ID\_auto"

INNER JOIN public."Model" m ON a."ID\_model" = m."ID\_model"

WHERE

j."Name" = 'Мастер-механик'

GROUP BY

e."ID\_employee",

m."Model"

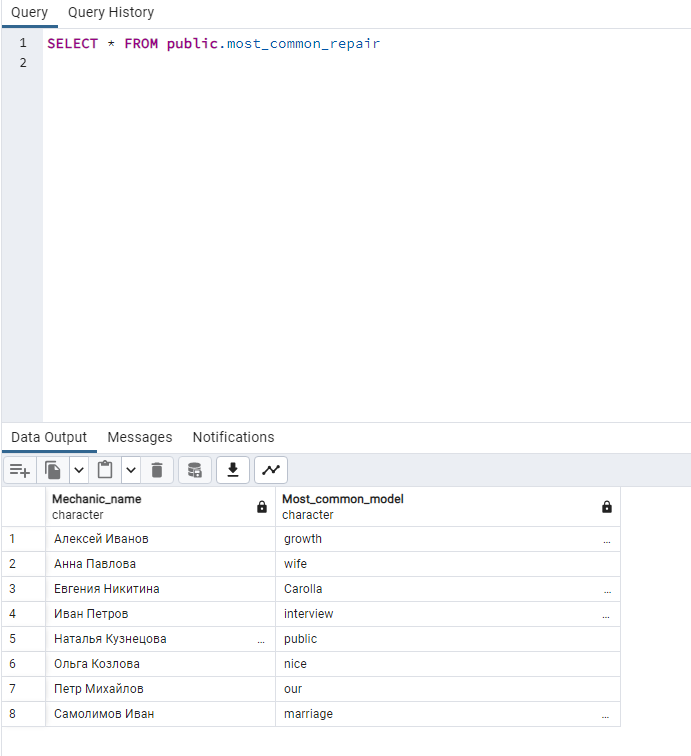
Order By

e."Full\_name"

) sub

WHERE

sub.rn = 1;



Для менеджеров (рассчитать премию все механикам, которые за прошедший месяц все свои заказы выполнили своевременно - 10% от зарплаты)

CREATE VIEW manager\_bonus AS

SELECT distinct

e."Full\_name" AS "Mechanic\_name",

j."Salary" \* 0.10 AS "Bonus"

FROM

public."Employee" e

INNER JOIN public."Job\_position" j ON e."ID\_job\_position" = j."ID\_job\_position"

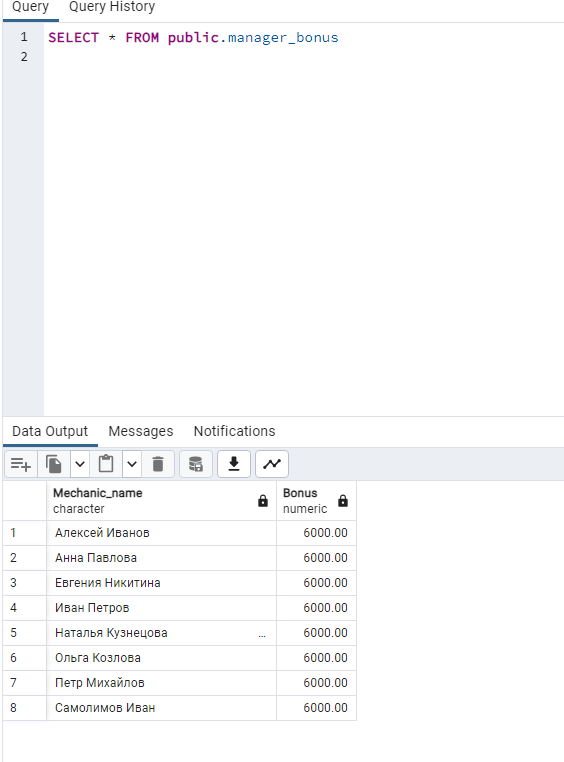
INNER JOIN public."Contract" c ON e."ID\_employee" = c."ID\_employee"

WHERE

j."Name" = 'Мастер-механик'

AND c."Actual\_date\_end\_of\_repair" BETWEEN date\_trunc('month', CURRENT\_DATE) AND date\_trunc('month', CURRENT\_DATE) + INTERVAL '1 month - 1 day'

AND c."Actual\_date\_end\_of\_repair" <= c."Scheduled\_date\_end\_of\_repair";



**Запросы на модификацию данных**

1) UPDATE запрос, обновляющий статус заказа для всех заказов, которые были приняты к ремонту и закончены в течение месяца.

UPDATE public."Contract" c1

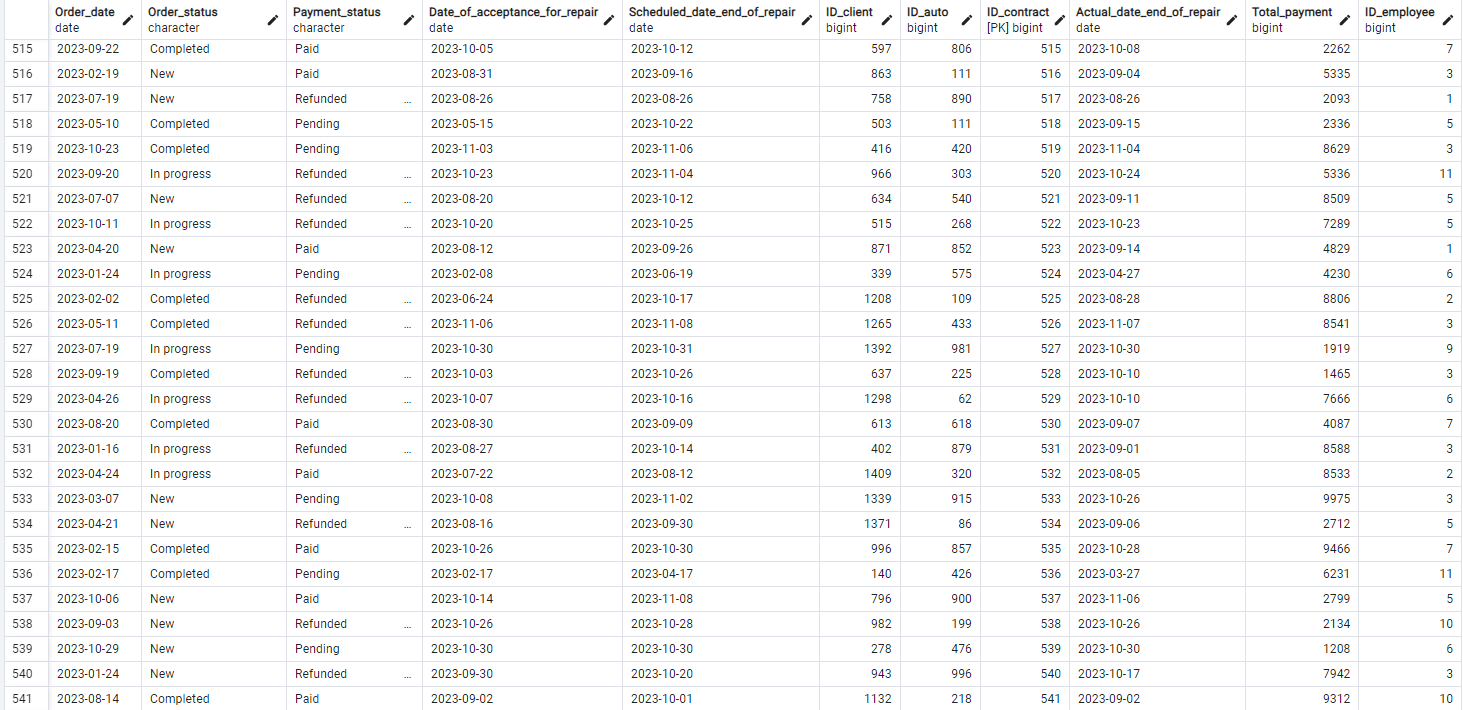
SET "Order\_status" = 'Completed'

WHERE "Order\_status" = 'In progress '

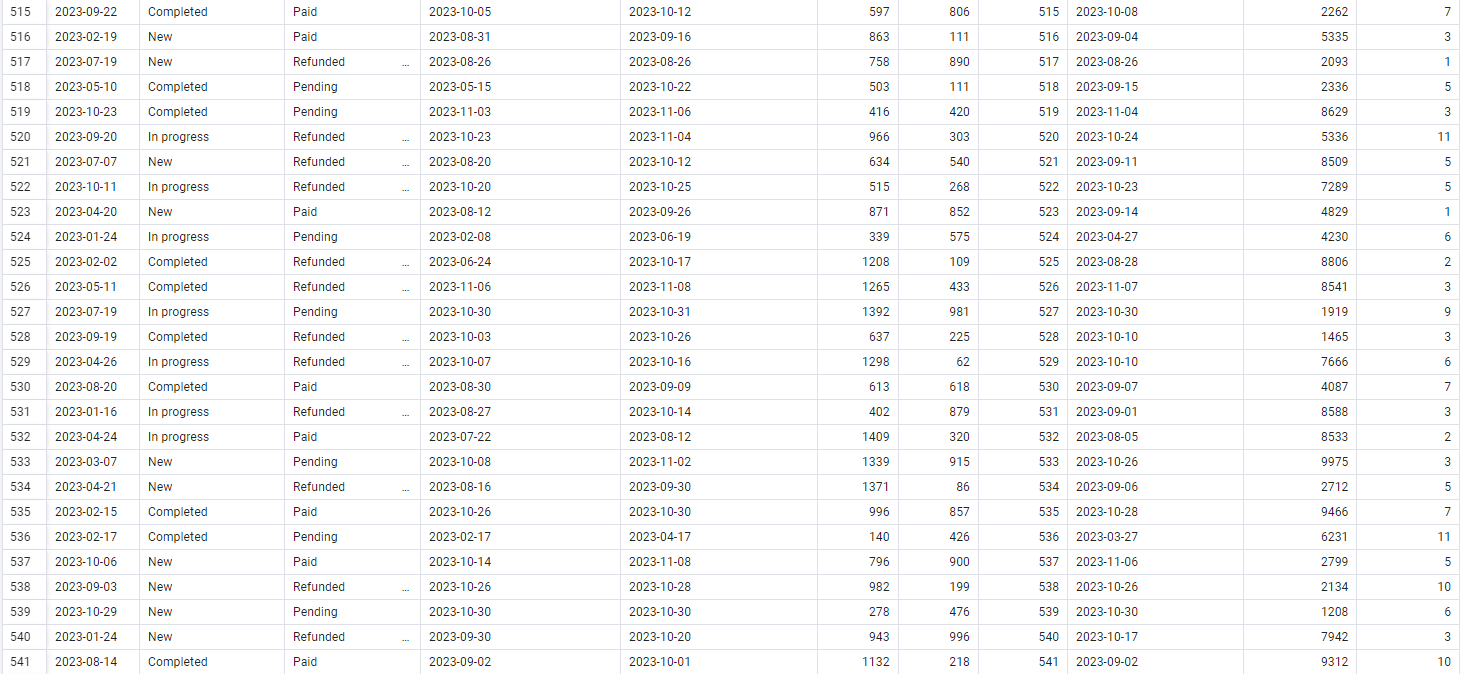
AND "Actual\_date\_end\_of\_repair" BETWEEN (SELECT date\_trunc('month', CURRENT\_DATE))

AND (SELECT date\_trunc('month', CURRENT\_DATE) + INTERVAL '1 month - 1 day');

До



После



2) INSERT запрос для вставки новых клиентов в таблицу Client для всех клиентов, которые уже имеют активные заказы:

INSERT INTO public."Client" ("Full\_name", "Phone", "Email", "ID\_client")

SELECT DISTINCT "Full\_name", "Phone", "Email", nextval('public."Client\_ID\_client\_seq"')

FROM public."Client" c

WHERE c."ID\_client" IN (

SELECT "ID\_client"

FROM public."Contract"

WHERE "Order\_status" = 'Accepted for repair'

);



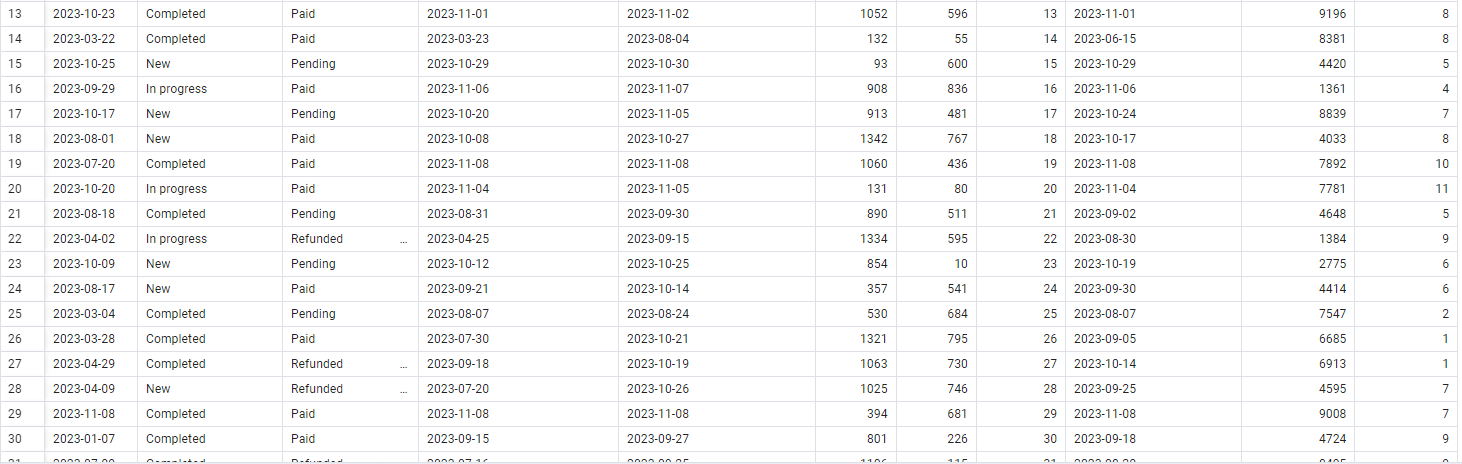
3) DELETE запрос, чтобы удалить все завершенные заказы, которые были приняты к ремонту и закончены в течение месяца:

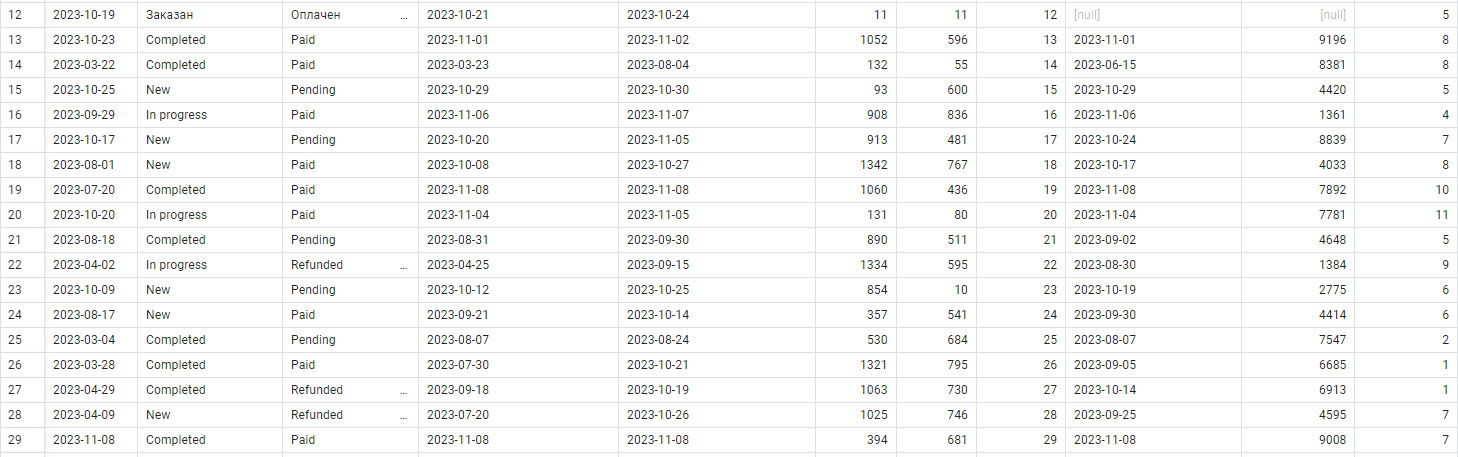
DELETE FROM public."Contract"

WHERE "Order\_status" = 'Paid'

AND "Actual\_date\_end\_of\_repair" BETWEEN (SELECT date\_trunc('month', CURRENT\_DATE))

AND (SELECT date\_trunc('month', CURRENT\_DATE) + INTERVAL '1 month - 1 day');





## 

## Вывод

В ходе проведения данной лабораторной работы были осуществлены разнообразные запросы к базе данных "Автомастерская", включая использование соединений таблиц, подзапросов и других методов. Также были выполнены запросы на создание представлений и модификацию данных, включая операции вставки, изменения и удаления.