

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
Факультет инфокоммуникационных технологий

ОТЧЁТ
О ЛАБОРАТОРНОЙ РАБОТЕ №2
Запросы на выборку и модификацию данных, представления и индексы в
PostgreSQL
по дисциплине:
ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ БАЗ ДАННЫХ

Выполнил: Пятыго Д. А.

Группа: K32421

Проверил: Говорова М. М.

Санкт-Петербург

2023

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Практическое задание:

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Схема базы данных: представлена на рисунке 1.



Рисунок 1. ER-диаграмма базы данных

Выполнение:

Запросы к базе данных:

- 1) Формулировка запроса: выбрать фамилию того механика, который чаще всех работает с автомобилями марки «Тойота».

Команда:

```

WITH Мастера_Тойота AS (
    SELECT
        "Сотрудник"."ФИО",
        COUNT(*) AS Количество_Тойот
    FROM
        "Сотрудник"
        JOIN "Должность" USING("ID_должности")
        JOIN "Состав_заказа" USING("Табельный_номер")
        JOIN "Заказ" USING("Номер_заказа")
        JOIN "Автомобиль" USING("Госномер")
    WHERE
        "Должность"."Название_должности" = 'Мастер'
        AND "Автомобиль"."Марка" = 'Toyota'
    GROUP BY
        "Сотрудник"."ФИО"
)

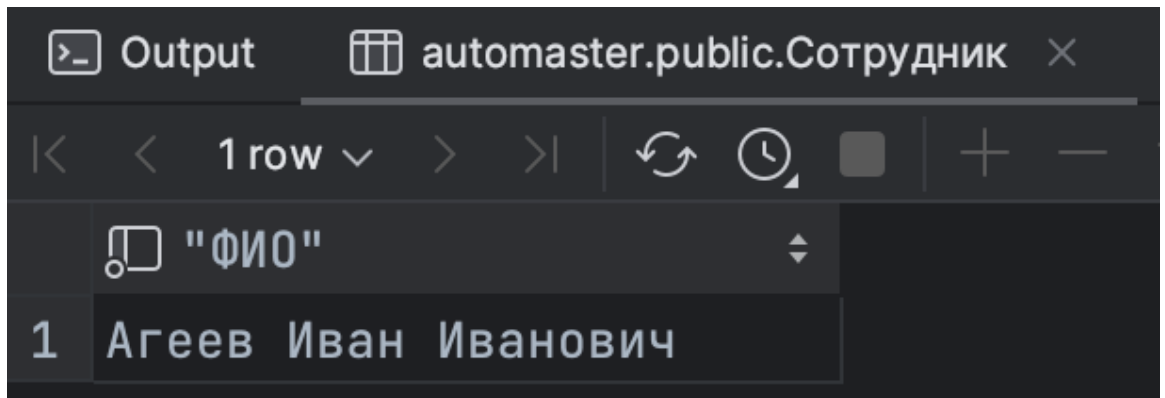
```

```

SELECT
    "ФИО"
FROM
    Мастера_Тойота
WHERE
    Количество_Тойот = (
        SELECT
            MAX(Количество_Тойот)
        FROM
            Мастера_Тойота
    );

```

Результат запроса: представлен на рисунке 2.



The screenshot shows a database query result viewer. At the top, there is a tab labeled "Output" and a window title "automaster.public.Сотрудник". Below the title bar, there is a toolbar with navigation icons (back, forward, first, last), a "1 row" indicator, and a refresh icon. The main area displays a table with one row. The first column is labeled "ФИО" and contains the value "Агеев Иван Иванович".

ФИО
Агеев Иван Иванович

Рисунок 2. Результат запроса 1

- 2) Формулировка запроса: определить тех владельцев автомобилей, которых всегда обслуживает один и тот же механик. Вывести фамилии механика и его постоянного клиента.

Команда:

```
WITH Клиент_Мастер AS (  
    SELECT DISTINCT  
        "Клиент"."ID_клиента",  
        "Клиент"."ФИО" AS ФИО_Клиент,  
        "Состав_заказа"."Табельный_номер",  
        "Сотрудник"."ФИО" AS ФИО_Сотрудник  
    FROM  
        "Сотрудник"  
    JOIN "Должность" USING("ID_должности")  
    JOIN "Состав_заказа" USING("Табельный_номер")  
    JOIN "Заказ" USING("Номер_заказа")  
    JOIN "Клиент" USING("ID_клиента")  
    WHERE  
        "Должность"."Название_должности" = 'Мастер'  
)  
Мастеров_на_клиента AS (  
    SELECT  
        Клиент_Мастер."ID_клиента",
```

```

COUNT(DISTINCT Клиент_Мастер."Табельный_номер") AS
Мастера
FROM
    Клиент_Мастер
GROUP BY
    Клиент_Мастер."ID_клиента"
)
SELECT
    Клиент_Мастер.ФИО_Клиент,
    Клиент_Мастер.ФИО_Сотрудник
FROM
    Клиент_Мастер
JOIN Мастеров_на_клиента ON Клиент_Мастер."ID_клиента" =
Мастеров_на_клиента."ID_клиента"
WHERE
    Мастеров_на_клиента.Мастера = 1

```

Результат запроса: представлен на рисунке 3.

	❏ ФИО_Клиент	❏ ФИО_Сотрудник
1	Белов Иван Александрович	Васин Дмитрий Александрович
2	Никитин Дмитрий Александрович	Агеев Иван Иванович

Рисунок 3. Результат запроса 2

- 3) Формулировка запроса: вывести фамилии механиков, которые не выполняли работы в срок и количество дней просрочки выполнения заказа.

Команда:

```

SELECT
    "Сотрудник"."ФИО",
    COALESCE("Состав_заказа"."Дата_исполнения", CURRENT_DATE)
- "Заказ"."Планируемая_дата_окончания" AS Дней_просрочки
FROM
    "Сотрудник"

```

```

JOIN "Должность" USING("ID_должности")
JOIN "Состав_заказа" USING("Табельный_номер")
JOIN "Заказ" USING("Номер_заказа")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND
    COALESCE("Состав_заказа"."Дата_исполнения", CURRENT_DATE)
> "Заказ"."Планируемая_дата_окончания"

```

Результат запроса: представлен на рисунке 4.

	☐ "ФИО"	☐ Дней_просрочки
1	Калинин Дмитрий Иванович	20
2	Васин Иван Иванович	20
3	Агеев Иван Иванович	8
4	Дорофеев Дмитрий Александрович	16
5	Агеев Сергей Александрович	16

Рисунок 4. Результат запроса 3

- 4) Формулировка запроса: вывести данные механика, который выполнял все виды ремонта за прошедшую неделю.

Команда:

```

WITH Мастера_Количество_Работ AS (
    SELECT
        "Сотрудник"."Табельный_номер",
        COUNT(DISTINCT "Состав_заказа"."ID_вида_работы") AS
Количество_работ
    FROM
        "Сотрудник"
        JOIN "Состав_заказа" USING("Табельный_номер")
    GROUP BY
        "Сотрудник"."Табельный_номер"
)
SELECT
    *

```

```

FROM
    "Сотрудник"
JOIN "Должность" USING("ID_должности")
JOIN "Состав_заказа" USING("Табельный_номер")
JOIN Мастера_Количество_Работ USING("Табельный_номер")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND
    "Состав_заказа"."Дата_исполнения" >= CURRENT_DATE -
INTERVAL '7 days' AND
    Мастера_Количество_Работ.Количество_работ = (SELECT COUNT(*)
FROM "Вид_работы")

```

Результат запроса: представлен на рисунке 5.

Табельный_номер	ID_должности	ФИО	ID_автомастерской	Заработная_плата	Название_должности
0 rows					

Рисунок 5. Результат запроса 4

- 5) Формулировка запроса: сколько заработал каждый мастер за прошедший месяц?

Команда:

```

SELECT
    "Сотрудник"."Табельный_номер",
    "Сотрудник"."ФИО",
    SUM("Вид_работы"."Стоимость_работы" * 0.5) AS "Зарплата"
FROM
    "Сотрудник"
JOIN "Должность" USING("ID_должности")
JOIN "Состав_заказа" USING("Табельный_номер")
JOIN "Вид_работы" USING("ID_вида_работы")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND

```



```

"Состав_заказа"."Статус_исполнения" = 'Завершён' AND
"Состав_заказа"."Дата_исполнения" >= CURRENT_DATE -
INTERVAL '1 month'
GROUP BY
"Сотрудник"."Табельный_номер"

```

Результат запроса: представлен на рисунке 6.

	□ "Табельный_номер" ▾	□ "ФИО" ▾	□ "Зарплата" ▾
1	4	Агеев Дмитрий Иванович	500
2	6	Агеев Иван Иванович	1000
3	9	Васин Дмитрий Александрович	2500

Рисунок 6. Результат запроса 5

- 6) Формулировка запроса: вывести данные владельцев автомобилей, которые обращались в ремонт больше одного раза.

Команда:

```

WITH Количество_Заказов AS (
    SELECT
        "ID_клиента",
        COUNT("Номер_заказа") AS Количество_заказов
    FROM
        "Заказ"
    GROUP BY
        "ID_клиента"
)
SELECT
    "Клиент".*
FROM
    "Клиент"
    JOIN Количество_Заказов USING("ID_клиента")
WHERE
    Количество_Заказов.Количество_заказов > 1

```

Результат запроса: представлен на рисунке 7.

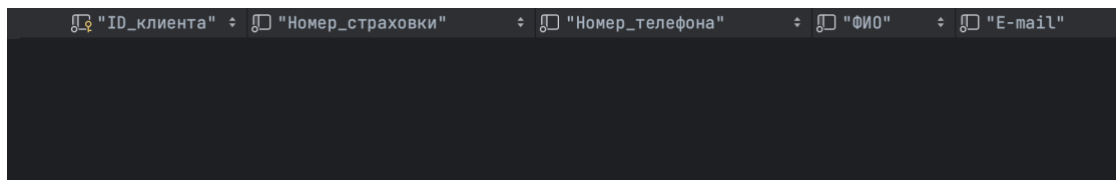


Рисунок 7. Результат запроса 6

- 7) Формулировка запроса: за каждый день просрочки выполнения заказа механику назначается штраф в размере 5%. Рассчитать штраф каждого механика за прошедший месяц.

Команда:

```
WITH Дней_Просрочки AS (
    SELECT
        "Сотрудник"."Табельный_номер",
        COALESCE("Состав_заказа"."Дата_исполнения",
CURRENT_DATE) - "Заказ"."Планируемая_дата_окончания" AS
Дней_просрочки
    FROM
        "Сотрудник"
    JOIN "Должность" USING("ID_должности")
    JOIN "Состав_заказа" USING("Табельный_номер")
    JOIN "Заказ" USING("Номер_заказа")
    WHERE
        "Должность"."Название_должности" = 'Мастер' AND
        COALESCE("Состав_заказа"."Дата_исполнения",
CURRENT_DATE) > "Заказ"."Планируемая_дата_окончания"
),
Зарплаты AS (
    SELECT
        "Сотрудник"."Табельный_номер",
        "Сотрудник"."ФИО",
        SUM("Вид_работы"."Стоимость_работы" * 0.5) AS Зарплата
    FROM
```

```

        "Сотрудник"
    JOIN "Должность" USING("ID_должности")
    JOIN "Состав_заказа" USING("Табельный_номер")
    JOIN "Вид_работы" USING("ID_вида_работы")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND
    "Состав_заказа"."Статус_исполнения" = 'Завершён' AND
    "Состав_заказа"."Дата_исполнения" >= CURRENT_DATE -
INTERVAL '1 month'
GROUP BY
    "Сотрудник"."Табельный_номер"
)
SELECT
    Зарплаты."Табельный_номер",
    Зарплаты."ФИО",
    Зарплаты.Зарплата * 0.05 * Дней_Просрочки.Дней_просрочки AS
Штраф
FROM
    Зарплаты
    JOIN Дней_Просрочки USING("Табельный_номер")

```

Результат запроса: представлен на рисунке 8.

	<input type="checkbox"/> "Табельный_номер" ▾	<input type="checkbox"/> "ФИО" ▾	<input type="checkbox"/> Штраф ▾
1	6	Агеев Иван Иванович	400

Рисунок 8. Результат запроса 7

Представления:

- 1) Формулировка запроса: для заказчиков (фамилию механика и модель автомобиля, которую он ремонтирует чаще всего).

Команда:

```

CREATE VIEW Для_заказчиков AS
WITH Модели_по_мастерам AS (

```

```

SELECT
    "Сотрудник"."Табельный_номер",
    "Сотрудник"."ФИО",
    "Автомобиль"."Марка",
    COUNT("Автомобиль"."Марка") AS Количество
FROM
    "Сотрудник"
    JOIN "Должность" USING("ID_должности")
    JOIN "Состав_заказа" USING("Табельный_номер")
    JOIN "Заказ" USING("Номер_заказа")
    JOIN "Автомобиль" USING("Госномер")
WHERE
    "Должность"."Название_должности" = 'Мастер'
GROUP BY
    "Сотрудник"."Табельный_номер",
    "Сотрудник"."ФИО",
    "Автомобиль"."Марка"
),
Макс_моделей_на_мастера AS (
    SELECT
        "Табельный_номер",
        MAX(Количество) AS Максимальное_количество
    FROM
        Модели_по_мастерам
    GROUP BY
        "Табельный_номер"
)
SELECT
    Модели_по_мастерам."ФИО",
    Модели_по_мастерам."Марка"

```

FROM

 Модели_по_мастерам

 JOIN Макс_моделей_на_мастера ON

Макс_моделей_на_мастера."Табельный_номер" =

Модели_по_мастерам."Табельный_номер" AND

Макс_моделей_на_мастера.Максимальное_количество =

Модели_по_мастерам.Количество

Результат запроса: представлен на рисунке 9.

```
SELECT
    Модели_по_мастерам."ФИО",
    Модели_по_мастерам."Марка"
FROM
    Модели_по_мастерам
JOIN Макс_моделей_на_мастера ON Макс_моделей_на_мастера."Табельный_номер" = Модели_по_мастерам.
[2023-04-05 22:14:44] completed in 54 ms
```

Рисунок 9. Результат выполнения запроса для создания представления 1

Содержимое представления: представлено на рисунке 10.

	❏ ФИО	❏ Марка
1	Агеев Дмитрий Иванович	Lada
2	Агеев Иван Иванович	Lada
3	Агеев Иван Иванович	Toyota
4	Агеев Сергей Александрович	Nissan
5	Васин Дмитрий Александрович	Volkswagen
6	Васин Иван Иванович	Lada
7	Дорофеев Дмитрий Александрович	Nissan
8	Калинин Дмитрий Иванович	Lada

Рисунок 10. Содержимое представления 1

- 2) Формулировка запроса: для менеджеров (рассчитать премию всем механикам, которые за прошедший месяц все свои заказы выполнили своевременно — 10% от зарплаты).

Команда:

CREATE VIEW Для_менеджеров AS

WITH Зарплаты_мастеров AS (

 SELECT

 "Сотрудник"."Табельный_номер",

```

        "Сотрудник"."ФИО",
        SUM("Вид_работы"."Стоимость_работы" * 0.5) AS "Зарплата"
FROM
    "Сотрудник"
    JOIN "Должность" USING("ID_должности")
    JOIN "Состав_заказа" USING("Табельный_номер")
    JOIN "Вид_работы" USING("ID_вида_работы")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND
    "Состав_заказа"."Статус_исполнения" = 'Завершён' AND
    "Состав_заказа"."Дата_исполнения" >= CURRENT_DATE -
INTERVAL '1 month'
GROUP BY
    "Сотрудник"."Табельный_номер"
),
Механики_Без_Премии AS (
    SELECT
        "Сотрудник"."Табельный_номер"
    FROM
        "Сотрудник"
        JOIN "Должность" USING("ID_должности")
        JOIN "Состав_заказа" USING("Табельный_номер")
        JOIN "Заказ" USING("Номер_заказа")
    WHERE
        "Должность"."Название_должности" = 'Мастер' AND
        COALESCE("Состав_заказа"."Дата_исполнения",
CURRENT_DATE) > "Заказ"."Планируемая_дата_окончания" AND
        COALESCE("Состав_заказа"."Дата_исполнения",
CURRENT_DATE) >= CURRENT_DATE - INTERVAL '1 month'
)

```

```

SELECT
    Зарплаты_мастеров."Табельный_номер",
    Зарплаты_мастеров."ФИО",
    Зарплаты_мастеров."Зарплата" * 0.1 AS "Премия"
FROM
    Зарплаты_мастеров
WHERE
    "Табельный_номер" NOT IN (SELECT "Табельный_номер" FROM
Механики_Без_Премии)

```

Результат запроса: представлен на рисунке 11.

```

Зарплаты_мастеров."Табельный_номер",
Зарплаты_мастеров."ФИО",
Зарплаты_мастеров."Зарплата" * 0.1 AS "Премия"
FROM
    Зарплаты_мастеров
WHERE
    "Табельный_номер" NOT IN (SELECT "Табельный_номер" FROM Механики_Без_Премии)
[2023-04-05 23:11:42] completed in 6 ms

```

Рисунок 11. Результат выполнения запроса для создания представления 2

Содержимое представления: представлено на рисунке 12.

	Табельный_номер	ФИО	Премия
1	4	Агеев Дмитрий Иванович	50
2	9	Васин Дмитрий Александрович	250

Рисунок 12. Содержимое представления 2

Запросы на модификацию данных:

- 1) Формулировка запроса: обновить заработную плату сотрудников в соответствии с объёмом выполненных работ за последний месяц.

Команда:

```

WITH Зарплаты_механиков AS (
    SELECT
        "Сотрудник"."Табельный_номер",
        SUM("Вид_работы"."Стоимость_работы" * 0.5) AS
"Заработная_плата"
    FROM
        "Сотрудник"

```

```

JOIN "Должность" USING("ID_должности")
JOIN "Состав_заказа" USING("Табельный_номер")
JOIN "Вид_работы" USING("ID_вида_работы")
WHERE
    "Должность"."Название_должности" = 'Мастер' AND
    "Состав_заказа"."Статус_исполнения" = 'Завершён' AND
    "Состав_заказа"."Дата_исполнения" >= CURRENT_DATE -
INTERVAL '1 month'
GROUP BY
    "Сотрудник"."Табельный_номер"
)
UPDATE
    "Сотрудник"
SET
    "Заработная_плата" = Зарплаты_механиков."Заработная_плата"
FROM
    Зарплаты_механиков
WHERE
    "Сотрудник"."Табельный_номер" =
Зарплаты_механиков."Табельный_номер"

```

Таблица до выполнения запроса: см. рисунок 13.

	WHERE		ORDER BY
	ФИО	ID_автомастерской	Заработная_плата
1	Антипов Андрей Викторович	1	70000
2	Кузьмин Аркадий Андреевич	2	70000
3	Агеев Дмитрий Александров...	1	0
4	Агеев Дмитрий Иванович	2	0
5	Агеев Иван Александрович	1	0
6	Агеев Иван Иванович	2	0
7	Агеев Сергей Александрович	1	0
8	Агеев Сергей Иванович	2	0
9	Васин Дмитрий Александров...	1	0
10	Васин Дмитрий Иванович	2	0
11	Васин Иван Александрович	1	0
12	Васин Иван Иванович	2	0
13	Васин Сергей Александрович	1	0
14	Васин Сергей Иванович	2	0

Рисунок 13. Таблица "Сотрудник" до выполнения запроса 1

Таблица после выполнения запроса: см. рисунок 14.

	ФИО	ID_автомастерской	Заработная_плата
13	Дорофеев Дмитрий Иванович	2	0
14	Дорофеев Иван Александров...	1	0
15	Дорофеев Иван Иванович	2	0
16	Дорофеев Сергей Александр...	1	0
17	Дорофеев Сергей Иванович	2	0
18	Калинин Дмитрий Александр...	1	0
19	Калинин Дмитрий Иванович	2	0
20	Калинин Иван Александрович	1	0
21	Калинин Иван Иванович	2	0
22	Калинин Сергей Александро...	1	0
23	Калинин Сергей Иванович	2	0
24	Агеев Дмитрий Иванович	2	500
25	Агеев Иван Иванович	2	1000
26	Васин Дмитрий Александров...	1	2500

Рисунок 14. Таблица "Сотрудник" после выполнения запроса 1

- 2) Формулировка запроса: добавить в базу данных заказ нового клиента Николаева Андрея Сергеевича (nikand@mail.ru, +79123456789, номер страховки АВ90580473) с серой Volvo 2005 года (Швеция) мощностью 150 лошадиных сил, госномер В153МХ78. Известно, что заказ оформил администратор Антипов Андрей Викторович. Планируемая дата окончания — неделя после текущей даты.

Команда:

```

INSERT INTO "Клиент" ("Номер_страховки", "Номер_телефона",
"ФИО", "E-mail")
VALUES ('AB90580473', '+79123456789', 'Николаев Андрей Сергеевич',
'nikand@mail.ru');

INSERT INTO "Автомобиль" ("Госномер", "Марка", "Мощность",
"Страна_производителя", "Год_выпуска", "Цвет", "ID_клиента")
VALUES ('B153MX78', 'Volvo', 150, 'Швеция', 2005, 'Серый', (
    SELECT "ID_клиента"
    FROM "Клиент"
    WHERE "ФИО" = 'Николаев Андрей Сергеевич'
    ORDER BY "ID_клиента" DESC
    LIMIT 1
));

INSERT INTO "Заказ" ("Дата_заказа", "Госномер", "ID_клиента",
"Табельный_номер", "Планируемая_дата_окончания")
VALUES (
    CURRENT_DATE,
    'B153MX78',
    (
        SELECT "ID_клиента"
        FROM "Клиент"
        WHERE "ФИО" = 'Николаев Андрей Сергеевич'
        ORDER BY "ID_клиента" DESC
        LIMIT 1
    ),
    (
        SELECT "Сотрудник"."Табельный_номер"
        FROM "Сотрудник"
        JOIN "Должность" USING("ID_должности")
        WHERE "Сотрудник"."ФИО" = 'Антипов Андрей Викторович' AND

```

```

        "Должность"."Название_должности" = 'Администратор'
    ),
    CURRENT_DATE + INTERVAL '7 days'
)

```

Таблица до выполнения запроса: см. рисунок 15.

	Номер_заказа	Дата_заказа	Госномер	ID_клиента	Табельный_номер
1	1	2023-02-10	M119HX33	1	1
2	2	2023-02-14	0069HC09	2	2
3	4	2023-01-28	H118CK43	4	2
4	5	2023-01-24	K270MB53	5	1
5	3	2023-02-12	A913CC66	3	1
6	6	2023-01-20	A5970P96	6	2

Рисунок 15. Таблица "Заказ" до выполнения запроса 2

Таблица после выполнения запроса: см. рисунок 16.

	Номер_заказа	Дата_заказа	Госномер	ID_клиента	Табельный_номер
1	1	2023-02-10	M119HX33	1	1
2	2	2023-02-14	0069HC09	2	2
3	4	2023-01-28	H118CK43	4	2
4	5	2023-01-24	K270MB53	5	1
5	3	2023-02-12	A913CC66	3	1
6	6	2023-01-20	A5970P96	6	2
7	7	2023-04-05	B153MX78	7	1

Рисунок 16. Таблица "Заказ" после выполнения запроса 2

- 3) Формулировка запроса: удалить всех клиентов, которые не делали заказы за последний год, при условии, что их автомобиль старше 15 лет. Гарантируется, что предварительно строки, ссылающиеся на номер заказа, удалены.

Команда:

```

WITH "Данные_для_удаления" AS (
    SELECT "Клиент"."ID_клиента",
           "Автомобиль"."Госномер"
    FROM "Клиент"
    JOIN "Автомобиль" USING("ID_клиента")
    WHERE "Автомобиль"."Год_выпуска" <= EXTRACT(YEAR FROM
CURRENT_DATE) - 15
    AND "Клиент"."ID_клиента" NOT IN (
        SELECT DISTINCT "Заказ"."ID_клиента"

```

```

FROM "Заказ"
WHERE "Заказ"."Дата_заказа" >= CURRENT_DATE - INTERVAL '1
year'
)
),
"Удалённые_заказы" AS (
DELETE FROM "Заказ"
WHERE "Госномер" IN (
SELECT "Госномер" FROM "Данные_для_удаления"
)
RETURNING "Госномер"
),
"Удалённые_автомобили" AS (
DELETE FROM "Автомобиль"
WHERE "Госномер" IN (
SELECT "Госномер" FROM "Удалённые_заказы"
)
RETURNING "ID_клиента"
)
DELETE FROM "Клиент"
WHERE "ID_клиента" IN (
SELECT "ID_клиента" FROM "Удалённые_автомобили"
)

```

Таблица до выполнения запроса: см. рисунок 17 (предварительно у Николаева Андрея Сергеевича изменили дату заказа на год раньше).

	ID_клиента	Номер_страховки	Номер_телефона	ФИО
1	1	J061179849	+79696231324	Белов Андрей Иванович
2	2	LB90312789	+79320990849	Никитин Андрей Александрович
3	3	IM64497399	+79813283946	Белов Иван Александрович
4	4	ID22670694	+79192908060	Никитин Дмитрий Александрович
5	5	LR18662028	+79396812195	Андреев Андрей Иванович
6	6	FG14429129	+79894558942	Никитин Андрей Иванович
7	7	AB90580473	+79123456789	Николаев Андрей Сергеевич

Рисунок 17. Таблица "Клиент" до выполнения запроса 3

Таблица после выполнения запроса: см. рисунок 18.

	ID_клиента	Номер_страховки	Номер_телефона	ФИО
1	1	J061179849	+79696231324	Белов Андрей Иванович
2	2	LB90312789	+79320990849	Никитин Андрей Александрович
3	3	IM64497399	+79813283946	Белов Иван Александрович
4	4	ID22670694	+79192908060	Никитин Дмитрий Александрович
5	5	LR18662028	+79396812195	Андреев Андрей Иванович
6	6	FG14429129	+79894558942	Никитин Андрей Иванович

Рисунок 18. Таблица "Клиент" после выполнения запроса 3

Создание индексов: время запроса будем вычислять с помощью EXPLAIN (ANALYZE, VERBOSE, BUFFERS) перед запросом. В качестве запроса возьмём запрос данных о мастерах IV разряда:

```
EXPLAIN (ANALYZE, VERBOSE, BUFFERS)
SELECT *
FROM "Сотрудник"
WHERE "ID_должности" IN (
    SELECT "ID_должности"
    FROM "Должность"
    WHERE "Название_должности" = 'Мастер' AND
        "Разряд" = 'IV'
)
```

Время запроса без индексов:

Planning Time: 0.270 ms

Execution Time: 0.123 ms

Время запроса с индексами (таблица «Сотрудник», столбцы «Табельный_номер», «ID_должности»):

Planning Time: 1.146 ms

Execution Time: 0.086 ms

С индексацией часто используемых столбцов (как, например, «ID_должности») время запроса сократилось в 2 раза.

Выводы: в данной лабораторной работе для спроектированной и заполненной данными базы я научился писать сложные запросы (как непосредственно к базе, так и на модификацию данных в ней), на практике изучив несколько новых полезных конструкций в SQL, создал два представления, а также сравнил скорость запросов с индексацией и без. Выяснилось, что индексация полезна для столбцов, которые участвуют в операциях фильтрации, сортировки или объединения — в таком случае, особенно при большом количестве данных, скорость выполнения запросов существенно возрастает.