# Министерство науки и высшего образования Российской Федерации ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский университет ИТМО» (Университет ИТМО)

Факультет прикладной информатики Образовательная программа Мобильные и сетевые технологии Направление подготовки 09.03.03 Мобильные и сетевые технологии

#### ОТЧЕТ

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

По дисциплине «Проектирование и реализация баз данных»

Автор: Турищев А. И.

Факультет: ИКТ Группа: K3340

Преподаватель: Говорова М. М.

# Содержание

1	Цель работы	2
2	Практическое задание	2
3	Схема базы данных	3
4	Выполнение         4.1 Запросы select          4.2 Представления          4.3 Запросы INSERT, DELETE, UPDATE          4.4 Индексы	4 10 11 15
5	Выводы	16

# 1 Цель работы

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

## 2 Практическое задание

#### Практическое задание:

- 1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию лабораторной работы №2, 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.

# 3 Схема базы данных

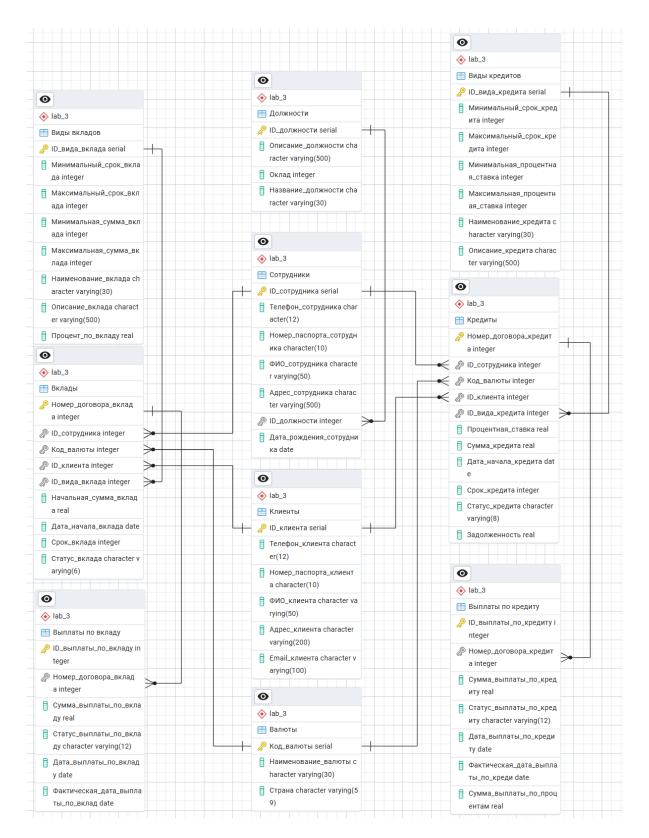


Рис. 1: ERD схема

## 4 Выполнение

#### 4.1 Запросы select

Запрос 1 - найти вкладчика, имеющего на текущий день несколько вкладов.

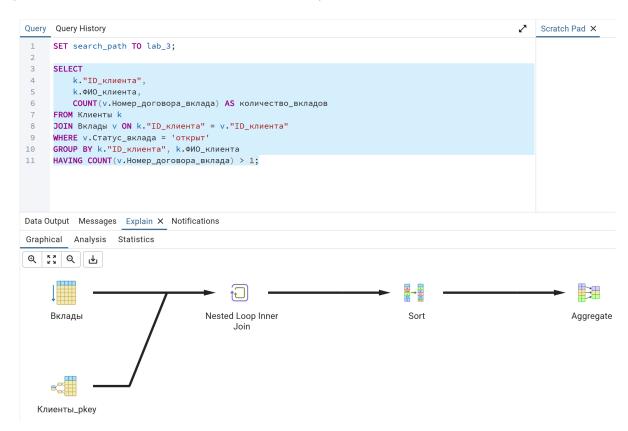


Рис. 2: Запрос 1

Запрос 2 - Найти вкладчика, имеющего вклады во нескольких видах валюты на текущий день.

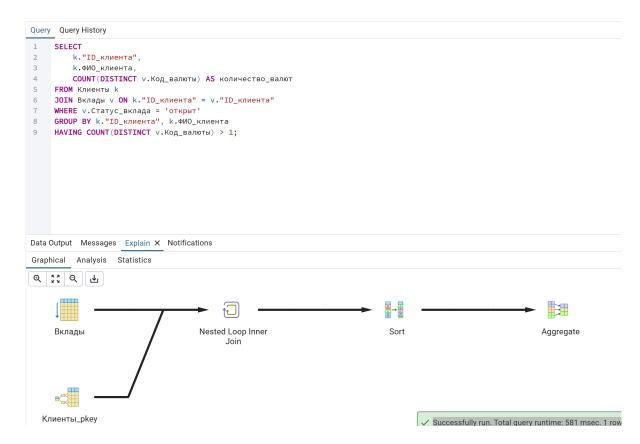


Рис. 3: Запрос 2

Запрос 3 - Вывести данные вкладчика, имеющего максимальный вклад в английских фунтах.

```
Query
       Query History
      SELECT
 1
 2
           k.*
      FROM Клиенты k
 3
       JOIN Вклады v ON k."ID_клиента" = v."ID_клиента"
 4
       JOIN Валюты val ON v.Код_валюты = val.Код_валюты
 5
      WHERE val. Наименование_валюты ILIKE '%фунт%'
 6
 7
      AND v.Статус_вклада = 'открыт'
      AND v. Начальная_сумма_вклада = (
 8
           SELECT MAX(Начальная_сумма_вклада)
 9
           FROM Вклады v2
10
           JOIN Валюты val2 ON v2.Код_валюты = val2.Код_валюты
11
12
           WHERE val2. Наименование_валюты ILIKE '%фунт%'
           AND v2.Статус_вклада = 'открыт'
13
14
       )
Data Output
            Messages
                        Explain X Notifications
Graphical
           Analysis
                      Statistics
⊕
          Q
               ♨
                        t 
  Вклады
                     Nested Loop Inner
                                            Aggregate
                                                                Nested Loop Inner
                        Join
                                                                   Join
  Валюты
                       Вклады
                                          Nested Loop Inner
                                             Join
                      Клиенты_pkey
```

Рис. 4: Запрос 3

Запрос 4 - Какой из вкладов пользовался наибольшей популярностью за истекший календарный год.

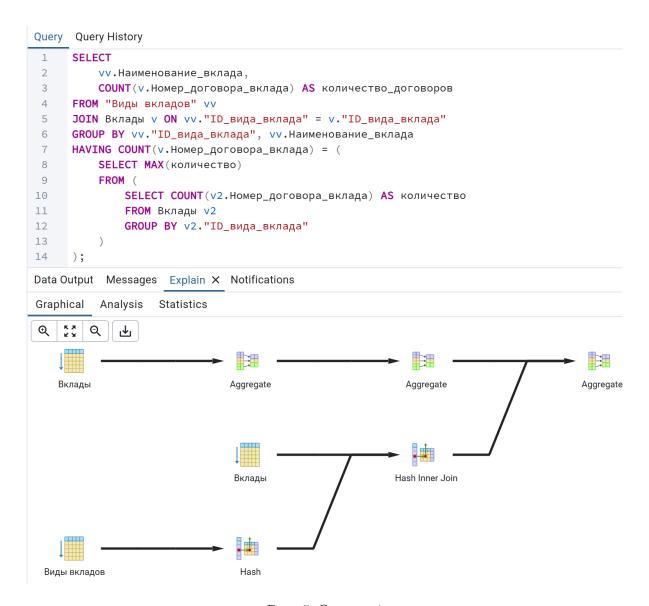


Рис. 5: Запрос 4

Запрос 5 - Кто из сотрудников заключил максимальное число договоров по кредитам за последний месяц.

#### Query History Query 1 **SELECT** 2 **s.**ФИО\_сотрудника, 3 COUNT(k. Номер\_договора\_кредита) AS количество\_кредитов 4 FROM Сотрудники s 5 JOIN Кредиты k ON s."ID\_сотрудника" = k."ID\_сотрудника" 6 WHERE k.Дата\_начала\_кредита >= CURRENT\_DATE - INTERVAL '1 month' 7 AND k.Дата\_начала\_кредита <= CURRENT\_DATE 8 GROUP BY s."ID\_сотрудника", s.ФИО\_сотрудника 9 **HAVING COUNT**(k.Номер\_договора\_кредита) = ( 10 SELECT MAX(макс\_количество) FROM ( 11 12 SELECT COUNT (k2. Номер\_договора\_кредита) AS макс\_количество 13 FROM Кредиты k2 14 WHERE k2.Дата\_начала\_кредита >= CURRENT\_DATE - INTERVAL '1 month' AND k2.Дата\_начала\_кредита <= CURRENT\_DATE 15 GROUP BY s."ID\_сотрудника", s.ФИО\_сотрудника 16 17 18 ) Data Output Messages Explain X Notifications Graphical Analysis Statistics Q € ♨ **1** Кредиты Nested Loop Inner Aggregate Join Сотрудники\_pkey Кредиты Aggregate Aggregate

Рис. 6: Запрос 5

Запрос 6 - Вывести список вкладчиков, у которых срок вклада истекает завтра и суммы начислений, которые могут быть ими востребованы.

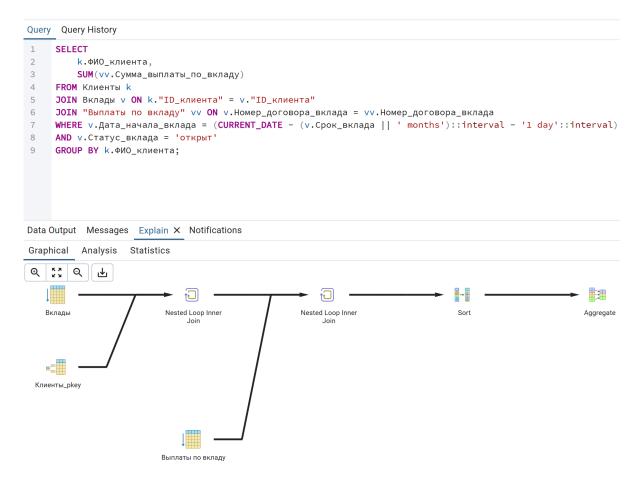


Рис. 7: Запрос 6

Запрос 7 - Вывести список сотрудников, заключивших договоры по вкладам на максимальную сумму за последний месяц.

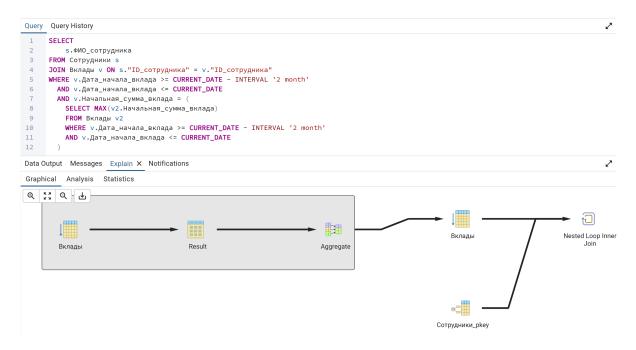


Рис. 8: Запрос 7

#### 4.2 Представления

Представление 1 - содержащее сведения обо всех сотрудниках банка и заключенных ими договорах по кредитам за прошедший месяц;

```
Сотрудники_кредиты_за_месяц
General Definition Code Security
      SELECT s."ID_сотрудника",
1
2
         s."ФИО_сотрудника",
         d."Название_должности",
3
4
         k."Номер_договора_кредита",
         k."Сумма_кредита",
         k."Процентная_ставка"
         k."Дата_начала_кредита"
         k."Дата_конца_кредита",
         k."Статус_кредита"
10
        FROM lab_3."Сотрудники" s
          JOIN lab_3."Должности" d ON s."ID_должности" = d."ID_должности"
11
          JOIN lab_3. "Кредиты" k ON s. "ID_сотрудника" = k. "ID_сотрудника"
12
       WHERE k."Дата_начала_кредита" >= (CURRENT_DATE - '1 mon'::interval) AND k."Дата_начала_кредита" <= CURRENT_DATE;
13
```

Рис. 9: Представление 1

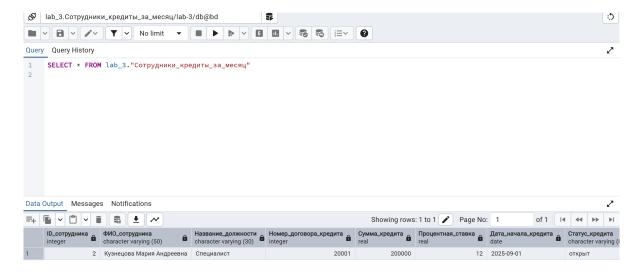


Рис. 10: Представление 1. просмотр

Представление 2 - найти клиентов банка, имеющих задолженности по кредитам.

Рис. 11: Представление 2

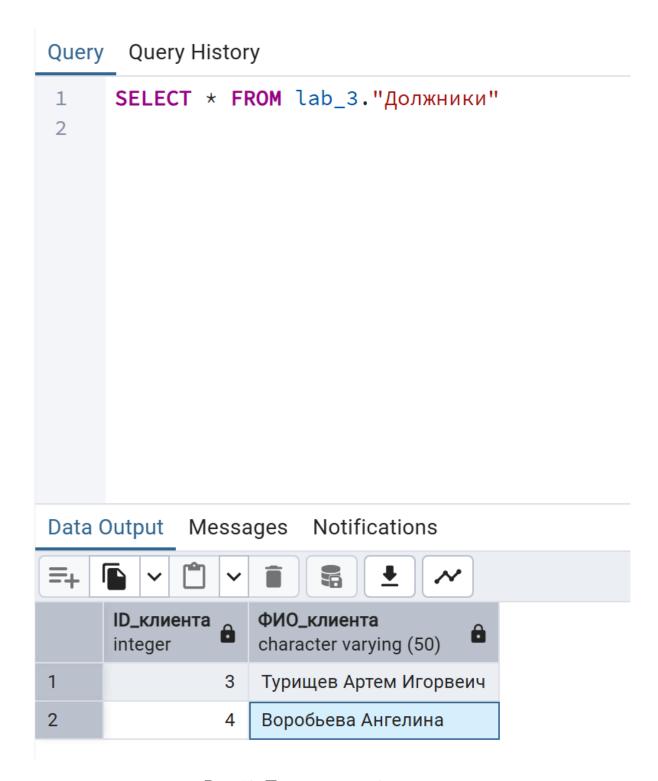


Рис. 12: Представление 2, просмотр

#### 4.3 Запросы INSERT, DELETE, UPDATE

INSERT

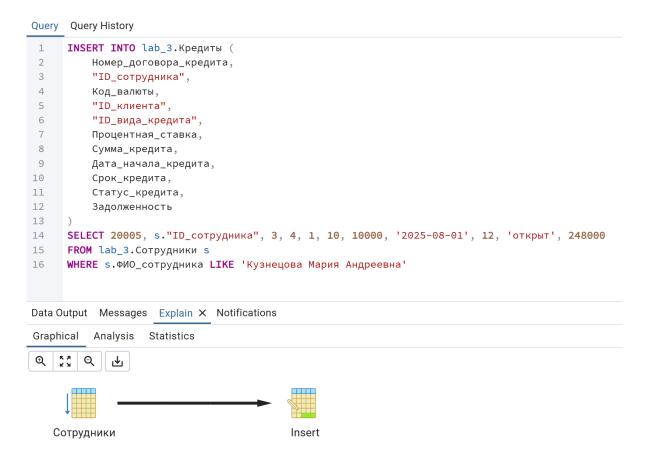


Рис. 13: Запрос INSERT

Был добавлен кредит с правильным id сотрудника(2)

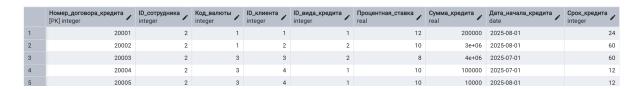


Рис. 14: Изменения запроса INSERT

#### UPDATE

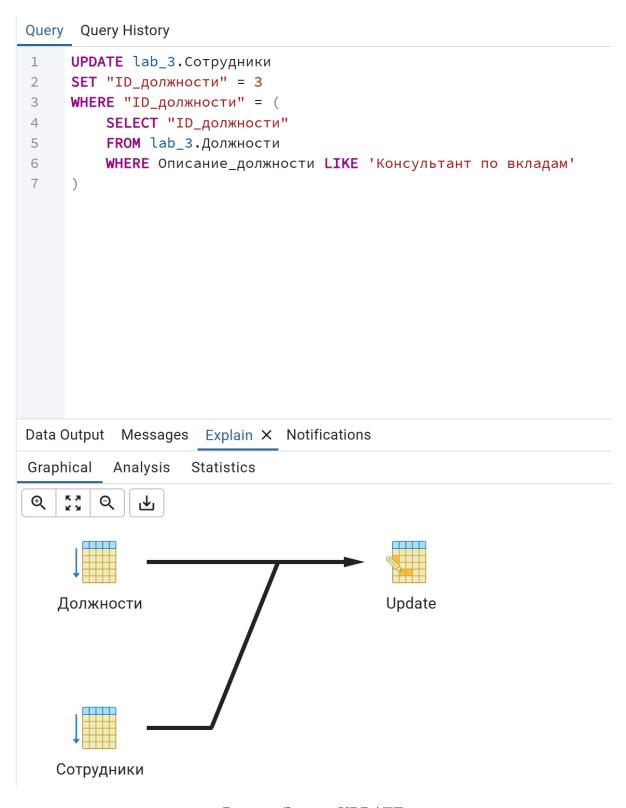


Рис. 15: Запрос UPDATE

Сотруднику был изменён ід должности на правильный(3)

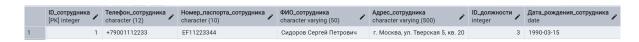


Рис. 16: Изменения запроса UPDATE

#### DELETE

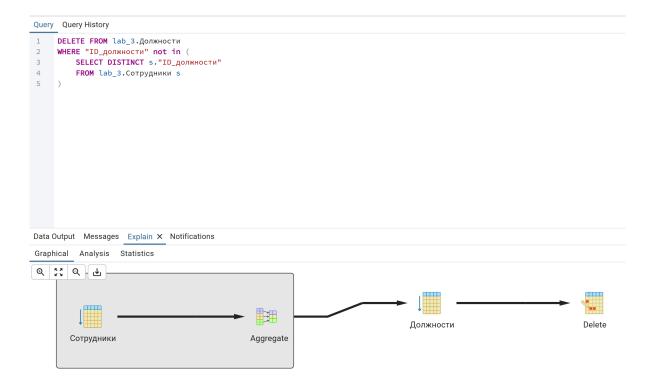


Рис. 17: Запрос DELET

Была удаленна должность 3, которую не имел ни один из сотрудников.

	<b>ID_должности</b> [PK] integer	Описание_должности character varying (500)	Оклад integer	Название_должности character varying (30)
1	1	Консультант по вкладам	25000	Консультант
2	2	Кредитный специалист	30000	Специалист

Рис. 18: Изменения запроса DELETE

#### 4.4 Индексы

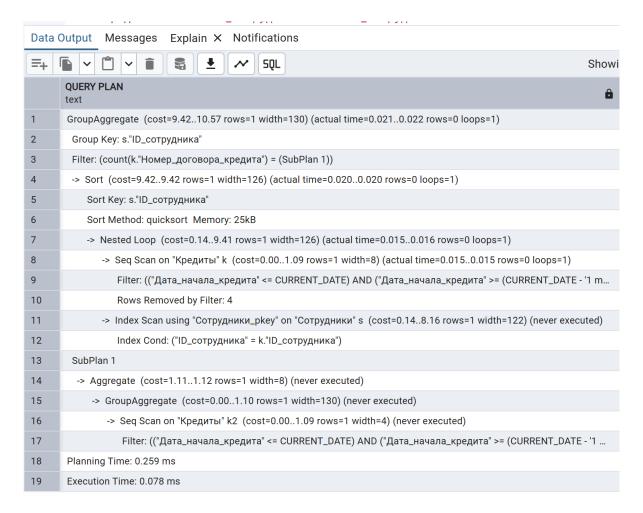


Рис. 19: До

	QUERY PLAN text
1	GroupAggregate (cost=9.4210.57 rows=1 width=130) (actual time=0.0140.015 rows=0 loops=1)
2	Group Key: s."ID_сотрудника"
3	Filter: (count(k."Номер_договора_кредита") = (SubPlan 1))
4	-> Sort (cost=9.429.42 rows=1 width=126) (actual time=0.0130.014 rows=0 loops=1)
5	Sort Key: s."ID_сотрудника"
6	Sort Method: quicksort Memory: 25kB
7	-> Nested Loop (cost=0.149.41 rows=1 width=126) (actual time=0.0090.010 rows=0 loops=1)
8	-> Seq Scan on "Кредиты" k (cost=0.001.09 rows=1 width=8) (actual time=0.0090.009 rows=0 loops=1)
9	Filter: (("Дата_начала_кредита" <= CURRENT_DATE) AND ("Дата_начала_кредита" >= (CURRENT_DATE - '1 m
10	Rows Removed by Filter: 4
11	-> Index Scan using "Сотрудники_pkey" on "Сотрудники" s (cost=0.148.16 rows=1 width=122) (never executed)
12	Index Cond: ("ID_сотрудника" = k."ID_сотрудника")
13	SubPlan 1
14	-> Aggregate (cost=1.111.12 rows=1 width=8) (never executed)
15	-> GroupAggregate (cost=0.001.10 rows=1 width=130) (never executed)
16	-> Seq Scan on "Кредиты" k2 (cost=0.001.09 rows=1 width=4) (never executed)
17	Filter: (("Дата_начала_кредита" <= CURRENT_DATE) AND ("Дата_начала_кредита" >= (CURRENT_DATE - '1
18	Planning Time: 0.224 ms
19	Execution Time: 0.045 ms

Рис. 20: После

# 5 Выводы

В ходе работы я овладел практическими навыками создания запросов на выборку и изменение данных с подзапросами, представлений, индексов и работы с ними в pgadmin и  $\operatorname{PostgreSQL}$