ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет инфокоммуникационных технологий

Дисциплина:

«Проектирование и реализация баз данных»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

«Запросы на выборку и модификацию данных, представления и индексы в PostgreSQL»

решолнил:
студент группы К33391
Черкес Артур Викторович
(подпись)
Проверил(а):
Говорова Марина Михайловна
(отметка о выполнении)
(nammar)

D.

Санкт-Петербург 2023

Т

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

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, pgadmin 4.

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.

Вариант 13. БД «Ресторан»

Описание предметной области: Необходимо создать систему для обслуживания заказов клиентов в ресторане.

Сотрудники ресторана – повара и официанты.

За каждым официантом закреплены определенные столы за смену. Клиенты могут бронировать столы заранее.

Каждый повар может готовить определенный набор блюд.

Официант принимает заказ от стола и передает его на кухню. Шеф-повар распределяет блюда для приготовления между поварами. В одном заказе может быть несколько одинаковых или разных блюд.

Запас продуктов на складе не должен быть ниже заданного значения.

Цена заказа складывается из стоимости ингредиентов и наценки, которая составляет 40% стоимости ингредиентов.

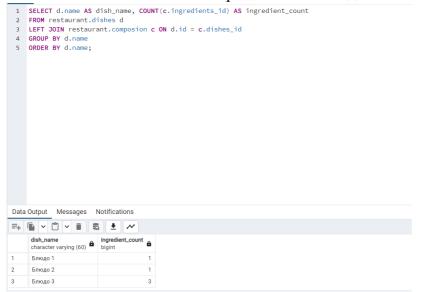
БД должна содержать следующий минимальный набор сведений: Табельный номер сотрудника. ФИО сотрудника. Паспортные данные сотрудника. Категория сотрудника. Должность сотрудника. Оклад сотрудника. Наименование ингредиента. Код ингредиента. Дата закупки. Объем закупки. Количество продукта на складе. Необходимый запас продукта. Срок годности. Цена ингредиента. Калорийность (на 100г продукта). Поставщик. Наименование

блюда. Код блюда. Объем ингредиента. Номер стола. Дата заказа. Код заказа. Количество. Название блюда. Ингредиенты, входящие в блюдо. Тип ингредиента.

Выполнение:

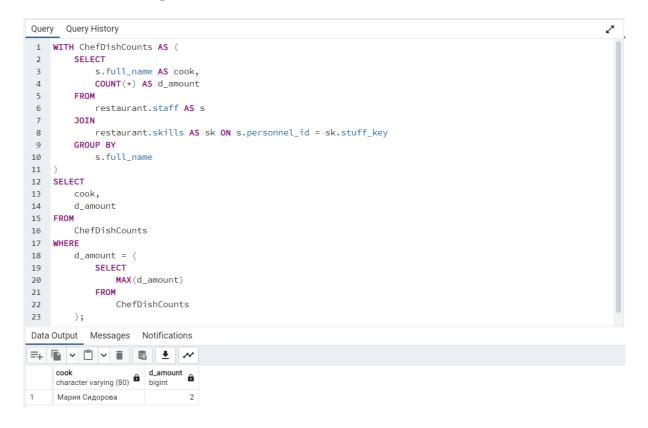
Запросы:

• Посчитать кол-во ингредиентов в блюде.

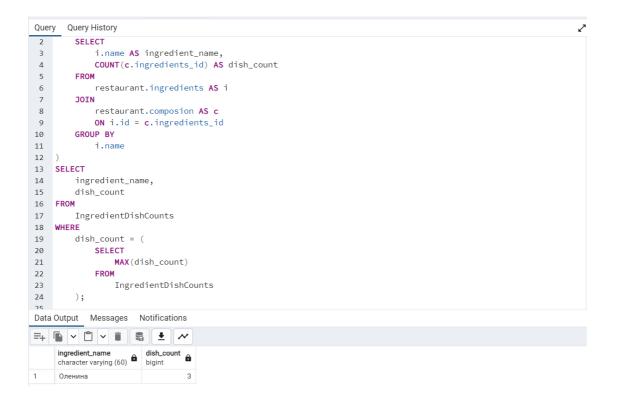


SELECT d.name AS dish_name, COUNT(c.ingredients_id) AS ingredient_count FROM restaurant.dishes d LEFT JOIN restaurant.composion c ON d.id = c.dishes_id GROUP BY d.name ORDER BY d.name;

Узнать, какой повар готовит больше всего блюд



Какой ингредиент содержится в максимальном количестве блюд



Количество блюд, заказанных в какой-то день:

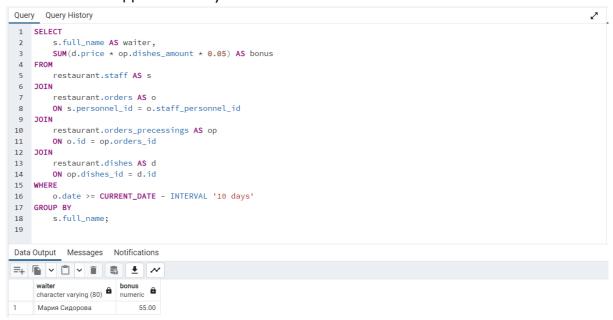
```
1 SELECT
2
        d.name AS dish_name,
        COUNT(*) AS dish_count
 4 FROM
 5 restaurant.orders AS o
 6 JOIN
7    restaurant.orders_precessings AS op
8    ON o.id = op.orders_id
9    JOIN
restaurant.dishes AS d
non.dishes_id = d.id
11
12 WHERE
13 o.date = '2023-10-02'
14 GROUP BY
d.name;
Data Output Messages Notifications
ingredient_name character varying (60) dish_count bigint
    Оленина
```

Какой повар приготовил больше всего блюд за смену

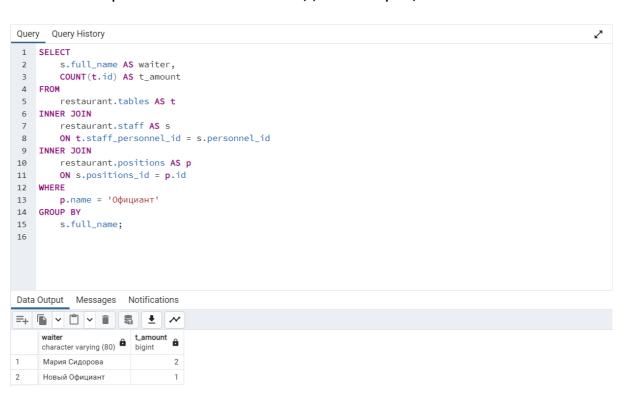
```
Query Query History
1 WITH CookDishCounts AS (
              SELECT
 2
                      d.staff_personnel_id AS cook_id,
 3
                     s.full_name AS cook_name,
 4
                     COUNT(*) AS d_amount
 5
 6
            FROM
 7
                     restaurant.dishes AS d
 8
            JOIN
 9
                     restaurant.staff AS s
                     ON d.staff_personnel_id = s.personnel_id
10
            JOIN
11
12
                     restaurant.orders_precessings AS op
                     ON op.dishes_id = d.id
13
14
            JOIN
15
                     restaurant.orders AS o
                     ON o.id = op.orders_id
16
           JOIN
17
18
                     restaurant.work_schedule AS ws
                     ON ws.staff_personnel_id = d.staff_personnel_id
19
20
            WHERE
21
                    ws.work_status = 'Рабочий'
22
23
24 )
                    d.staff_personnel_id, s.full_name
25 SELECT
            cook_id,
27
              cook_name,
28
             d_amount
30
            CookDishCounts
31 WHERE
            d_amount = (SELECT MAX(d_amount) FROM CookDishCounts);
33
Data Output Messages Notifications

    □
    ∨
    □
    √
    □
    √
    □
    ✓
    □
    ✓
    □
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓
    ✓</
                                                          d_amount
        1 Иван Иванов
                     2 Петр Петров
```

Рассчитать премию каждого официанта за последние 10 дней (5% от стоимости каждого заказа).

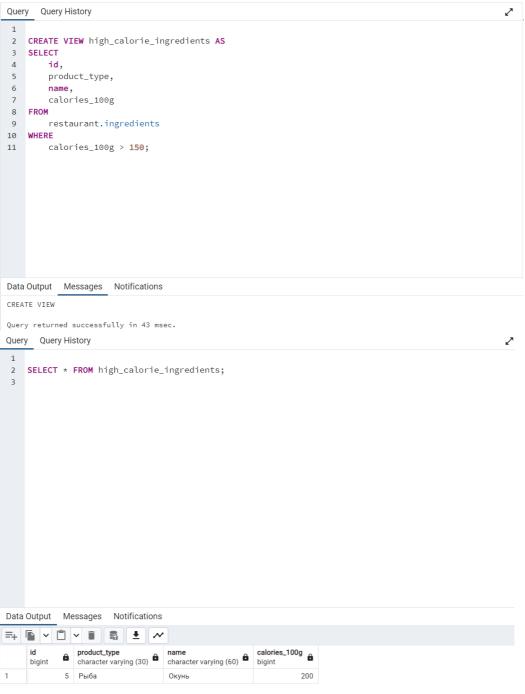


Сколько закреплено столов за каждым из официантов?



Представления:

Список ингредиентов с калорийностью более 150 калорий на 100 грамм.



Представление с информацией о сотрудниках и их графике работы:

```
CREATE VIEW staff_work_schedule AS
  SELECT
                        s.full_name AS staff_name,
                       ws.work_status AS work_status,
                       ss.beginning AS shift_start,
                       ss.ending AS shift_end
  FROM
                        staff s
  JOIN
                       work_schedule ws ON s.personnel_id = ws.staff_personn
  JOIN
                        shifts_schedule ss ON ws.shift_id = ss.id;
Messages
Query returned successfully in 47 msec.
   Query Query History
    1 SELECT * FROM staff_work_schedule;
   Data Output Messages Notifications
  staff_name character varying (80) a work_status character varying (20) a shift_start timestamp without time zone a shift_end timestamp without timestamp without timestamp without timestamp without timestamp
                                                           Рабочий
               Иван Иванов
                                                                                                                2023-10-02 00:00:00
                                                                                                                                                                               2023-10-02 11:59:59
              Петр Петров
                                                               Рабочий
                                                                                                                 2023-10-02 12:00:00
                                                                                                                                                                              2023-10-02 23:59:59
```

2023-10-02 12:00:00

2023-10-02 23:59:59

Модификация данных

Отсутствует

Мария Сидорова

```
Query Query History

1 UPDATE restaurant.orders
2 SET date = '2023-10-05', status = 'Изменен', payment_state = 'He оплачен'
3 WHERE id = (SELECT id FROM restaurant.orders WHERE payment_state = 'He оплачен' LIMIT 1);

Data Output Messages Notifications

UPDATE 1
```

Удаление заказа, еду для которого приготовил повар с самой высокой ставкой

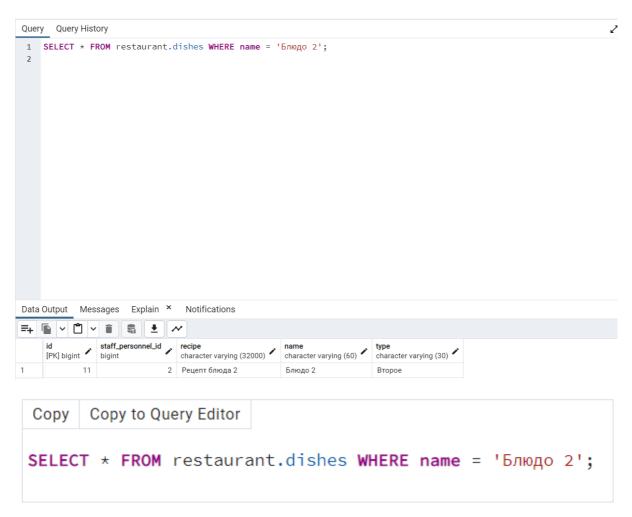
```
DELETE FROM restaurant.orders_precessings
WHERE orders_id IN (
    SELECT id
    FROM restaurant.orders
    WHERE staff_personnel_id = (
        SELECT personnel_id
        FROM restaurant.staff
        WHERE positions_id = (
            SELECT positions_id
            FROM restaurant.positions
            ORDER BY working_rates DESC
LIMIT 1)
LIMIT 1);
```

Messages

Query returned successfully in 67 msec.

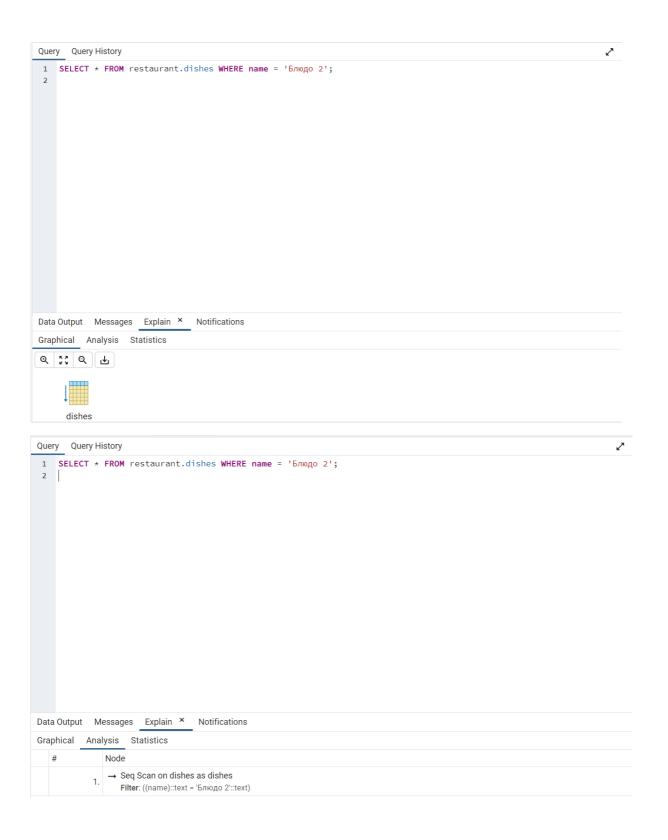
Создание индексов

Без индекса:



Messages

Successfully run. Total query runtime: 60 msec. 1 rows affected.



Создание индекса:

```
Copy Copy to Query Editor

CREATE INDEX idx_dishes_name ON restaurant.dishes (name);
```

Messages

Query returned successfully in 39 msec.

Поиск с индексом:

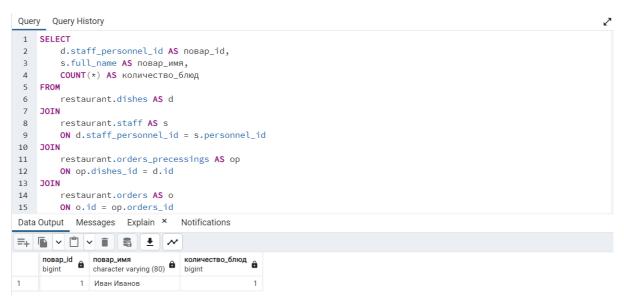
```
SELECT *
FROM restaurant.dishes
WHERE name = 'Блюдо 2';
```

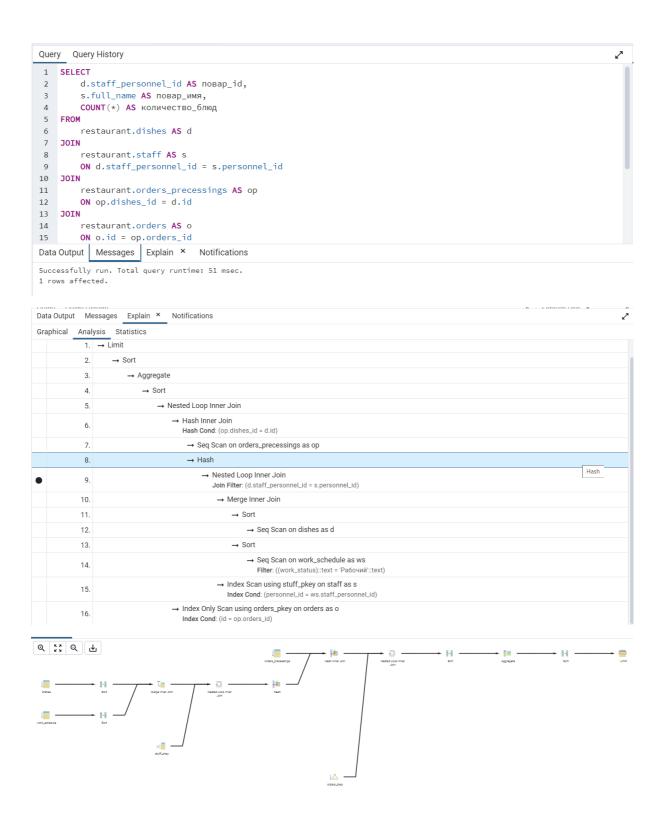
Messages

Successfully run. Total query runtime: 53 msec. 1 rows affected.

Оказался быстрее на 7 мс

Без индекса:





Создание индекса:



Поиск и индексом:

```
Query History
6
        restaurant.dishes AS d
8
     restaurant.staff AS s
ON d.staff_personnel_id = s.personnel_id
10 JOIN
restaurant.orders_precessings AS op

ON op.dishes_id = d.id
13 JOIN
     restaurant.orders AS o
14
15 ON
16 JOIN
        ON o.id = op.orders_id
17    restaurant.work_schedule AS ws
18    ON ws.staff_personnel_id = d.staff_personnel_id
19 WHERE
20
       ws.work_status = 'Рабочий'
21 GROUP BY
22 d.sta
23 ORDER BY
      d.staff_personnel_id, s.full_name
      количество_блюд DESC
25 LIMIT 1;
Data Output | Messages | Explain × Notifications
Successfully run. Total query runtime: 45 msec.
```

Быстрее на 6 мс

Индексы помогают сократить время сложного запроса, но на примере простых запросов мы видим, что планировщик считает что лучше просканировать обычным способом

Выводы

SQL запросы позволяют изменять, добавлять или удалять данные, а также составлять различные выборки, подсчитывать числовые характеристики. Сравнив время выполнения запросов с индексами и без, можно сделать вывод, что с индексами запросы выполнялись примерно столько же. Это связано с небольшим количеством данных в таблице.