

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение высшего  
образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**О Т Ч Е Т**

по лабораторной работе

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

по дисциплине «Проектирование и реализация баз данных»

Автор: Глушков Кирилл Георгиевич

Факультет: ИКТ

Группа: К32422

Преподаватель: Говорова Марина Михайловна

Дата сдачи: 15.04.23



Санкт-Петербург 2023

**СОДЕРЖАНИЕ**

1 Описание работы.....	2
2 Схема базы данных .....	2
3 Выполнение запросов .....	3

## 1 Описание работы

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.

Индивидуальное задание по варианту «Аэропорт».

## 2 Схема базы данных

Схема была создана с использованием генератора ERD-схемы в pgadmin.

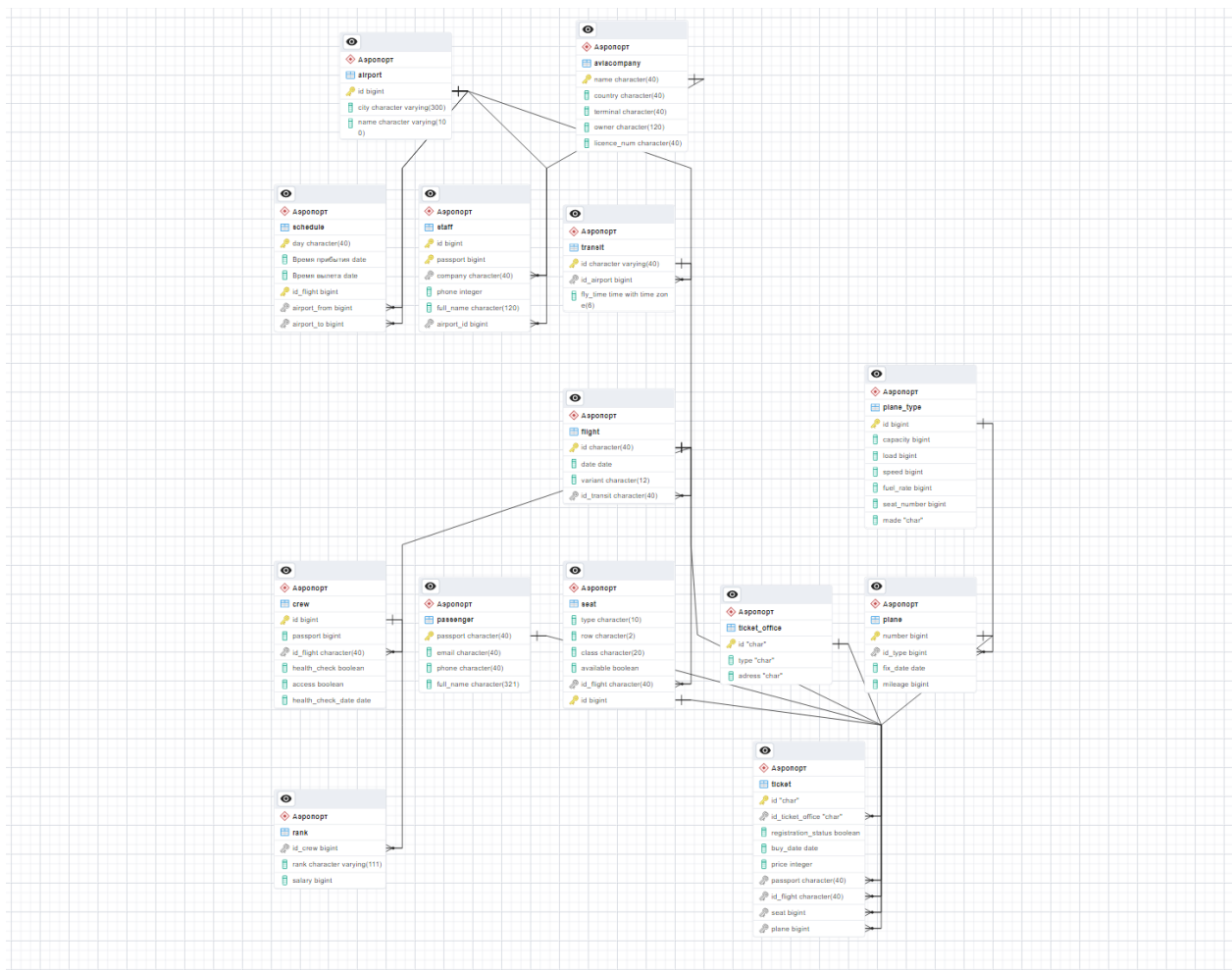


Рисунок 1 – Схема базы данных «Аэропорт»

### 3 Выполнение запросов

1. Для извлечения всех строк из таблицы `airport`:

```
```sql
SELECT * FROM Аэропорт.airport;
```
```

2. Для получения всех аэропортов в городе "Москва":

```
```sql
SELECT * FROM Аэропорт.airport WHERE city = 'Москва';
```
```

3. Чтобы добавить новый аэропорт с именем "name 10" в городе "Москва":

```
```sql
```

```
INSERT INTO Аэропорт.airport (city, name) VALUES ('Москва', 'name 10');
```

```
```
```

4. Чтобы обновить имя аэропорта с id=3 на "new name":

```
```sql
```

```
UPDATE Аэропорт.airport SET name = 'new name' WHERE id = 3;
```

```
```
```

5. Чтобы удалить все аэропорты, находящиеся в городе "Саратов":

```
```sql
```

```
DELETE FROM Аэропорт.airport WHERE city = 'Саратов';
```

```
```
```

1. Получить все записи из таблицы Аэропорт.transit:

```
```sql
```

```
SELECT * FROM Аэропорт.transit;
```

```
```
```

2. Получить все записи, где id\_airport больше 5:

```
```sql
```

```
SELECT * FROM Аэропорт.transit WHERE id_airport > 5;
```

```
```
```

3. Обновить время вылета для записи с конкретным id (допустим, мы хотим обновить запись с id = 3):

```
```sql
```

```
UPDATE Аэропорт.transit SET fly_time = '13:30' WHERE id = 3;
```

```
---
```

4. Вставить новую запись в таблицу Аэропорт.transit:

```
```sql
```

```
INSERT INTO Аэропорт.transit (id, id_airport, fly_time) VALUES (11, 10, '15:00');
```

```
---
```

5. Удалить запись из таблицы Аэропорт.transit (допустим, id записи, которую мы хотим удалить, равен 7):

```
```sql
```

```
DELETE FROM Аэропорт.transit WHERE id = 7;
```

```
---
```

1. Получить все записи из таблицы Аэропорт.flight:

```
```sql
```

```
SELECT * FROM Аэропорт.flight;
```

```
---
```

2. Получить записи, где variant = 'home':

```
```sql
```

```
SELECT * FROM Аэропорт.flight WHERE variant = 'home';
```

```
---
```

3. Обновить значение поля variant для записи с конкретным id (допустим, мы хотим обновить запись с id = 3):

```
```sql
```

```
UPDATE Аэропорт.flight SET variant = 'away' WHERE id = 3;
```

```
---
```

4. Вставить новую запись в таблицу Аэропорт.flight:

```
```sql
```

```
INSERT INTO Аэропорт.flight (id, date, variant, id_transit) VALUES (11, '2021-10-01', 'home', '11');
```

```
---
```

5. Удалить запись из таблицы Аэропорт.flight (допустим, id записи, которую мы хотим удалить, равен 7):

```
```sql
```

```
DELETE FROM Аэропорт.flight WHERE id = 7;
```

```
---
```

1. Получить все записи из таблицы Аэропорт.crew:

```
```sql
```

```
SELECT * FROM Аэропорт.crew;
```

```
---
```

2. Получить записи, где health\_check = True и access = False:

```
```sql
```

```
SELECT * FROM Аэропорт.crew WHERE health_check = TRUE AND access = FALSE;
```

```
---
```

3. Обновить значение поля access для записей, где id\_flight = 5:

```
```sql
```

```
UPDATE Аэропорт.crew SET access = TRUE WHERE id_flight = '5';
```

```
---
```

4. Вставить новую запись в таблицу Аэропорт.crew:

```
```sql
```

```
INSERT INTO Аэропорт.crew (id, passport, id_flight, health_check, access, health_check_date)  
VALUES (11, '254970805739', '11', TRUE, FALSE, '2021-10-01');
```

```
---
```

5. Удалить запись из таблицы Аэропорт.crew (допустим, id записи, которую мы хотим удалить, равен 7):

```
```sql
```

```
DELETE FROM Аэропорт.crew WHERE id = 7;
```

```
```
```

6. Получить количество записей в таблице Аэропорт.crew:

```
```sql
```

```
SELECT COUNT(*) FROM Аэропорт.crew;
```

```
```
```

7. Получить запись с максимальным значением поля id:

```
```sql
```

```
SELECT * FROM Аэропорт.crew ORDER BY id DESC LIMIT 1;
```

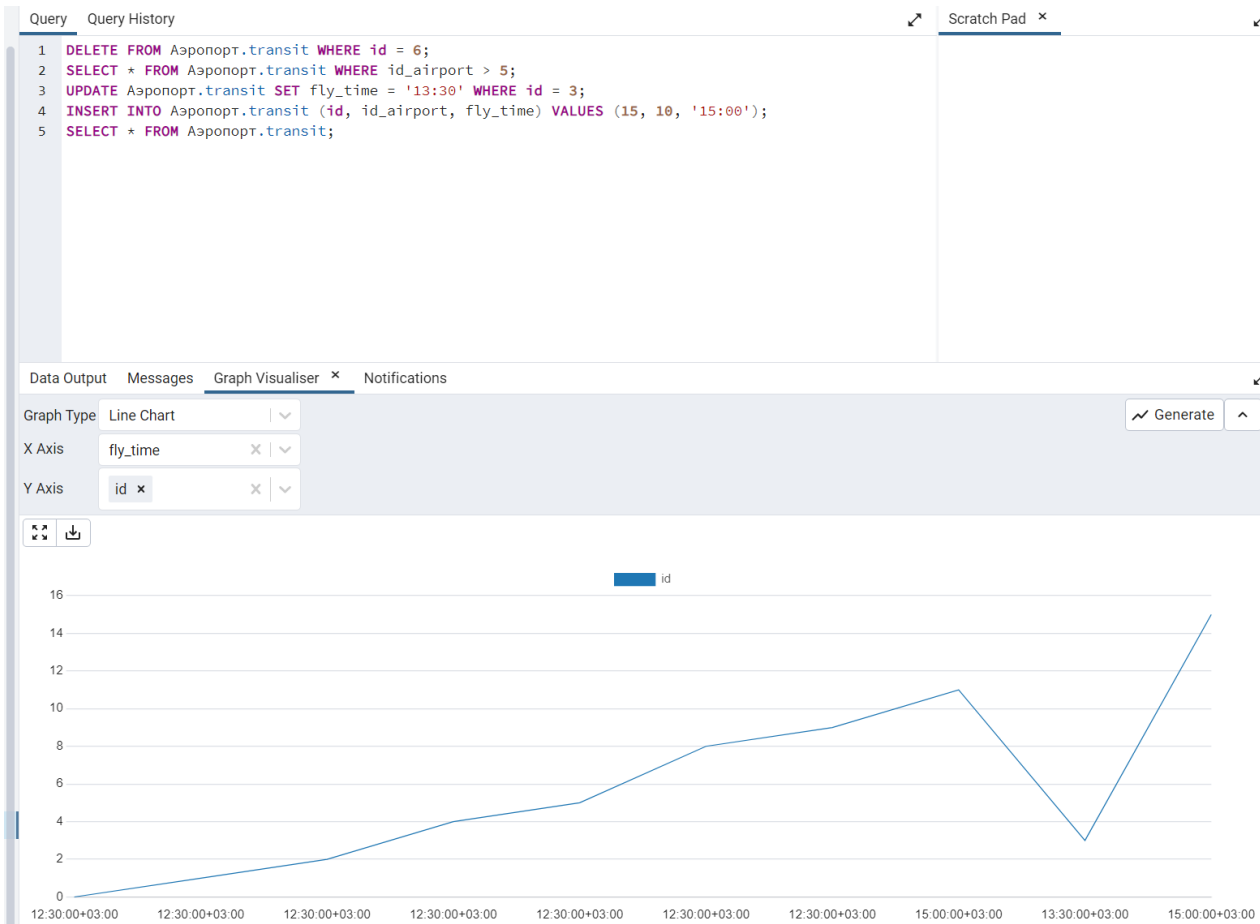
```
```
```

8. Получить записи с health\_check\_date в промежутке между двумя датами (например, между '2021-10-01' и '2021-10-10'):

```
```sql
```

```
SELECT * FROM Аэропорт.crew WHERE health_check_date BETWEEN '2021-10-01' AND '2021-10-10';
```

```
```
```





Query
Query History

```

1 DELETE FROM Аэропорт.transit WHERE id = 6;
2 SELECT * FROM Аэропорт.transit WHERE id_airport > 5;
3 UPDATE Аэропорт.transit SET fly_time = '13:30' WHERE id = 3;
4 INSERT INTO Аэропорт.transit (id, id_airport, fly_time) VALUES (126, 10, '15:00');
5 SELECT * FROM Аэропорт.transit;

```

Data Output
Messages
Notifications

|    | id<br>[PK] bigint | id_airport<br>bigint | fly_time<br>time with time zone |
|----|-------------------|----------------------|---------------------------------|
| 1  | 0                 | 1                    | 12:30:00+03:00                  |
| 2  | 1                 | 2                    | 12:30:00+03:00                  |
| 3  | 2                 | 3                    | 12:30:00+03:00                  |
| 4  | 4                 | 5                    | 12:30:00+03:00                  |
| 5  | 5                 | 6                    | 12:30:00+03:00                  |
| 6  | 8                 | 9                    | 12:30:00+03:00                  |
| 7  | 9                 | 10                   | 12:30:00+03:00                  |
| 8  | 11                | 10                   | 15:00:00+03:00                  |
| 9  | 15                | 10                   | 15:00:00+03:00                  |
| 10 | 16                | 10                   | 15:00:00+03:00                  |
| 11 | 116               | 10                   | 15:00:00+03:00                  |
| 12 | 1126              | 10                   | 15:00:00+03:00                  |
| 13 | 3                 | 4                    | 13:30:00+03:00                  |
| 14 | 126               | 10                   | 15:00:00+03:00                  |

## ЗАКЛЮЧЕНИЕ

В рамках лабораторной работы были созданы запросы и представления на выборку данных к базе данных PostgreSQL согласно индивидуальному заданию, часть 2 и 3. Были созданы 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов. Были изучены графические представления запросов. Были созданы простой и составной индексы для двух произвольных запросов.

Таким образом, за выполнение данной лабораторной работы удалось познакомиться с представлениями и индексами и успешно их реализовать.

Также были отработаны навыки выполнения запросов на индивидуальных заданиях. Успешно были реализованы различные модификации данных с подзапросами.

Индексы при больших запросах позволили значительно выиграть время выполнения, план запроса остался тем же.