

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: Будунов Б.С.

Факультет: ИКТ

Группа: K3241

Преподаватель: Говорова М.М.



Санкт-Петербург 2024

Оглавление

Цель работы	3
Практическое задание	3
Индивидуальное задание	3
Результаты выполнения команд.....	5
Запросы на вставку, удаление и изменение	8
Представления	11
Запросы с индексами и без	13
Вывод.....	15

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.

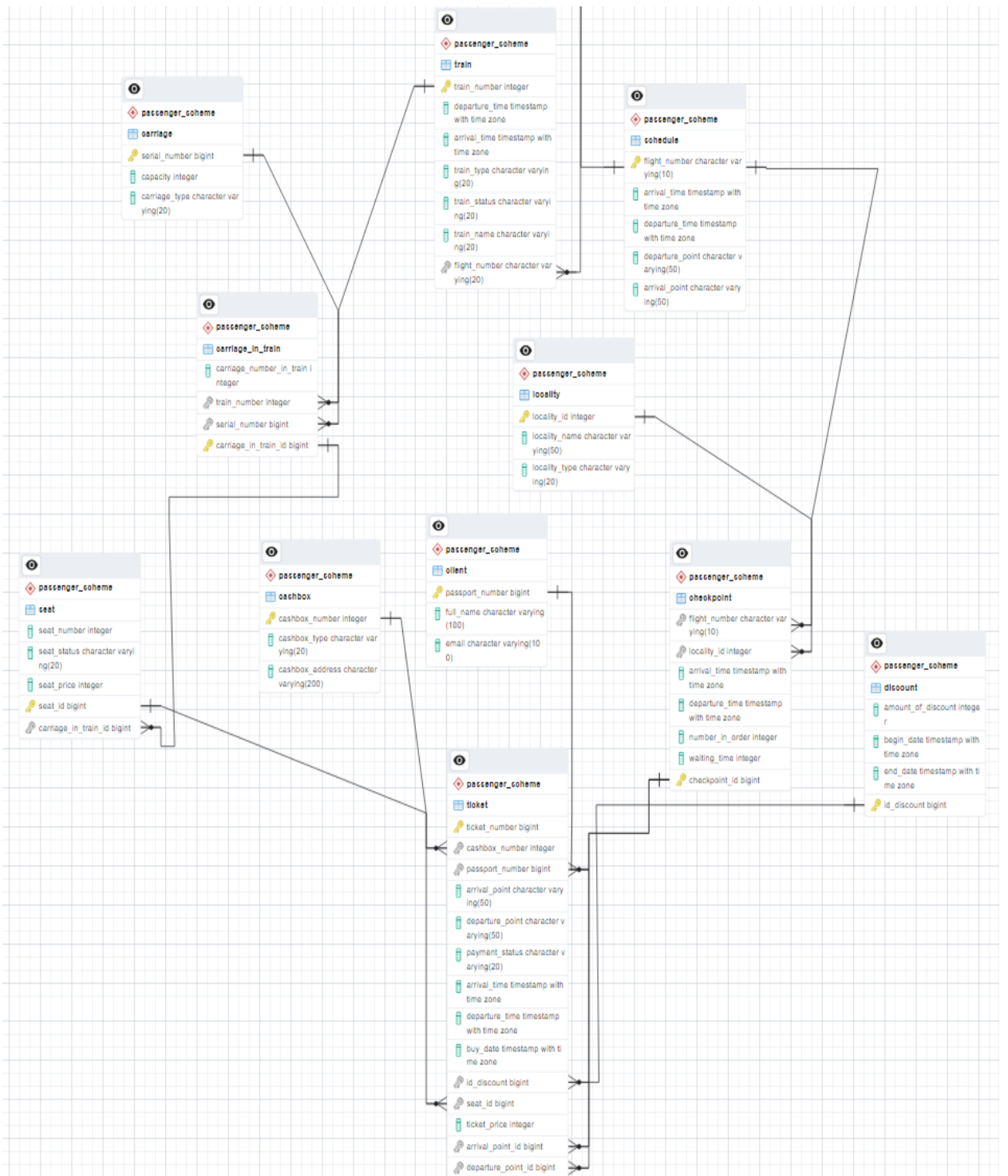
Индивидуальное задание:**Задание 2.** Создать запросы:

- Свободные места на все поезда, отправляющиеся с вокзала в течение следующих суток.
- Список пассажиров, отправившихся в Москву всеми рейсами за прошедшие сутки.
- Номера поездов, на которые проданы все билеты на следующие сутки.
- Свободные места в купейные вагоны всех рейсов до Москвы на текущие сутки.
- Выручка от продажи билетов на все поезда за прошедшие сутки.
- Общее количество билетов, проданных по всем направлениям в вагоны типа “СВ”.
- Номера и названия поездов, все вагоны которых были заполнены менее чем наполовину за прошедшие сутки.

Задание 3. Создать представление:

- для пассажиров о наличии свободных мест на заданный рейс;
- количество непроданных билетов на все поезда, формирующиеся за прошедшие сутки (номер поезда, тип вагона, количество).

Схема базы данных:



Команда 1

Активация Windows

Чтобы активировать Windows, перейдите в раздел "Параметры".

- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (11)
 - carriage
 - carriage_in_train
 - cashbox
 - checkpoint
 - client
 - discount
 - locality
 - schedule
 - seat
 - ticket
 - train**
- Trigger Functions
- Types
- Views
 - public
- Subscriptions

Активация Windows
Чтобы активировать Windows, перейдите в

5

Query Query History

```

1 SELECT t.train_number
2 FROM passenger_scheme.seat s, passenger_scheme.carriage_in_train cit,
3 passenger_scheme.train t
4 WHERE s.carriage_in_train_id = cit.carriage_in_train_id
5 AND cit.train_number = t.train_number
6 AND t.departure_time BETWEEN '2024-01-18 00:00:00' AND '2024-01-19 00:00:00'
7 GROUP BY t.train_number
8 HAVING COUNT(*) = SUM(CASE WHEN s.seat_status = 'Куплено' THEN 1 ELSE 0 END);

```

Messages Notifications Data Output

train_number [PK] integer

1	7
---	---

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 1 of 1 Query complete 00:00:00.117 Ln 5, Col 38

Команда 4

passenger/postgres@PostgreSQL 16

Query Query History

```

1 SELECT s.seat_id, s.seat_number, cit.carriage_number_in_train, t.train_number, sch.flight_number
2 FROM passenger_scheme.seat s, passenger_scheme.carriage_in_train cit, passenger_scheme.carriage c,
3 passenger_scheme.train t, passenger_scheme.schedule sch
4 WHERE s.carriage_in_train_id = cit.carriage_in_train_id AND cit.train_number = t.train_number
5 AND t.flight_number = sch.flight_number AND cit.serial_number = c.serial_number AND c.carriage_type = 'Kyne'
6 AND s.seat_status = 'Свободно' AND sch.arrival_point = 'Москва'
7 AND sch.departure_time BETWEEN '2024-01-17 00:00:00' AND '2024-01-18 00:00:00'

```

Messages Notifications Data Output

	seat_id bigint	seat_number integer	carriage_number_in_train integer	train_number integer	flight_number character varying (10)
2	30	3	1	5	flt_2
3	42	4	1	5	flt_2
4	57	5	2	5	flt_2
5	54	5	1	5	flt_2
6	21	2	2	5	flt_2
7	34	3	2	5	flt_2
8	46	4	2	5	flt_2
9	60	6	1	5	flt_2
10	63	6	2	5	flt_2

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 10 of 10 Query complete 00:00:00.097 Ln 7, Col 1

Команда 5

Query Query History

```

1 SELECT SUM(t.ticket_price) AS income
2 FROM passenger_scheme.ticket t
3 WHERE t.buy_date BETWEEN '2024-01-16 00:00:00' AND '2024-01-17 00:00:00';

```

Messages Notifications Data Output

	income bigint
1	33000

Команда 6

QueryQuery History

1SELECT COUNT(s.seat_id)

2FROM passenger_scheme.seat s, passenger_scheme.carriage_in_train cit, passenger_scheme.carriage c

3WHERE s.carriage_in_train_id = cit.carriage_in_train_id AND cit.serial_number = c.serial_number

4AND s.seat_status = 'Куплено' AND c.carriage_type = 'CB'

MessagesNotificationsData Output

count

bigint

1

15

Активация Windows
Чтобы активировать Windows, перейдите в

Команда 7

QueryQuery History

1SELECT subquery.train_number, subquery.train_name

2FROM (

3SELECT t.train_number, t.train_name, c.carriage_number_in_train,

4COUNT(*) AS total_seats,

5SUM(CASE WHEN s.seat_status = 'Куплено' THEN 1 ELSE 0 END) AS taken_seats,

6SUM(CASE WHEN s.seat_status = 'Куплено' THEN 1 ELSE 0 END) * 100.0 / COUNT(*) AS occupancy_percentage

7FROM passenger_scheme.train t

8JOIN passenger_scheme.carriage_in_train c ON t.train_number = c.train_number

9JOIN passenger_scheme.seat s ON c.carriage_in_train_id = s.carriage_in_train_id

10WHERE t.departure_time BETWEEN '2024-01-17 00:00' AND '2024-01-18 00:00'

11GROUP BY t.train_number, t.train_name, c.carriage_number_in_train

12) AS subquery

13WHERE occupancy_percentage < 50

14GROUP BY subquery.train_number, subquery.train_name

MessagesNotificationsData Output

train_number

train_name

[PK] integer

character varying (20)

1

5

Поезд_02

Активация Windows
Чтобы активировать Windows, перейдите в

Запросы на вставку, удаление и изменение
Вставка

Query

Query History

1

INSERT INTO passenger_scheme.ticket

2

VALUES(DEFAULT,

3

(SELECT cashbox_number FROM passenger_scheme.cashbox WHERE cashbox_type = 'онлайн'),

4

(SELECT passport_number FROM passenger_scheme.client WHERE full_name = 'Будунов Будун'),

5

'Москва', 'Санкт-Петербург', 'Оплачено',

6

'2024-01-26 05:00:00', '2024-01-25 17:45:00', '2024-01-24', 1,

7

(SELECT s.seat_id FROM passenger_scheme.seat s

8

JOIN passenger_scheme.carriage_in_train cit ON cit.carriage_in_train_id = s.carriage_in_train_id

9

JOIN passenger_scheme.train t ON t.train_number = cit.train_number

10

WHERE s.seat_status = 'Свободно' AND t.train_type = 'Спец. назначения'), 0)

Messages

Notifications

Data Output

INSERT 0 1

Query returned successfully in 82 msec.

Активация Windows

Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 0 of 0

Query complete 00:00:00.082

Ln 7, Col 50

Query

Query History

1

SELECT * FROM passenger_scheme.ticket

2

ORDER BY ticket_number ASC

Messages

Notifications

Data Output

	ticket_number [PK] bigint	cashbox_number integer	passport_number bigint	arrival_point character varying (50)	departure_point character varying (50)	payment_status character varying (20)	arrival_time timestamp with time zone
25	30	3	8217004	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
26	31	3	8217005	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
27	32	3	8217006	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
28	33	3	8217007	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
29	34	3	8217008	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
30	35	3	8217008	Москва	Санкт-Петербург	Опалчено	2024-01-18 18:00:00
31	36	3	8217001	Москва	Санкт-Петербург	Опалчено	2024-01-17 18:00:00
32	37	3	8217002	Москва	Санкт-Петербург	Опалчено	2024-01-17 18:00:00
33	38	3	8217003	Москва	Санкт-Петербург	Опалчено	2024-01-17 18:00:00
34	39	3	8217956767	Москва	Санкт-Петербург	Оплачено	2024-01-26 05:00:00
35	40	3	8217956767	Москва	Санкт-Петербург	Оплачено	2024-01-26 05:00:00

uses > passenger > Schemas > passenger scheme > Tables > ticket

Ln 1, Col 1

Удаление

Query Query History

```

1 DELETE FROM passenger_scheme.seat
2 WHERE (carriage_in_train_id) IN (
3     SELECT cit.carriage_in_train_id
4     FROM passenger_scheme.carriage_in_train cit
5     JOIN passenger_scheme.train t ON cit.train_number = t.train_number
6     JOIN passenger_scheme.carriage c ON cit.serial_number = c.serial_number
7     WHERE (c.carriage_type = 'Плацкарт' AND cit.carriage_number_in_train = 1) OR t.train_number < 4
8 )

```

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Messages Notifications Data Output

ases > passenger > Schemas > passenger_scheme > Tables > seat Ln 7, Col 96

Query Query History

```

1 SELECT train.train_number FROM passenger_scheme.train
2 WHERE train.train_number IN (
3     SELECT t.train_number FROM passenger_scheme.train t, passenger_scheme.carriage_in_train cit,
4     passenger_scheme.carriage c
5     WHERE t.train_number = cit.train_number AND c.serial_number = cit.serial_number
6     AND (c.carriage_type = 'Плацкарт' AND cit.carriage_number_in_train = 1)
7     OR t.train_number < 4
8 )

```

Data Output Messages Notifications

train_number	
[PK] integer	
1	3

Query Query History

```

1 SELECT seat.train_number, seat.car_number_in_train FROM passenger_scheme.seat
2 WHERE (seat.train_number, seat.car_number_in_train) IN (
3     SELECT t.train_number, cit.carriage_number_in_train
4     FROM passenger_scheme.train t, passenger_scheme.carriage_in_train cit,
5     passenger_scheme.carriage c
6     WHERE t.train_number = cit.train_number AND c.serial_number = cit.serial_number
7     AND (c.carriage_type = 'Плацкарт' AND cit.carriage_number_in_train = 1)
8     OR t.train_number < 4
9 )

```

Data Output Messages Notifications

train_number	car_number_in_train
integer	integer

Изменения

Query Query History

```

1 UPDATE passenger_scheme.seat
2 SET seat_status = 'Куплено'
3 WHERE car_number_in_train IN (
4     SELECT carriage_number_in_train
5     FROM passenger_scheme.carriage_in_train
6     WHERE train_number IN (
7         SELECT train_number
8         FROM passenger_scheme.train
9         WHERE train_type = 'Спец. назначения' AND departure_time < '2024-01-26 00:00:00'
10    )
11 )
12 AND train_number IN (
13     SELECT train_number
14     FROM passenger_scheme.train
15     WHERE train_type = 'Спец. назначения' AND departure_time < '2024-01-26 00:00:00'
16 )
17 AND seat_status = 'Свободно';
18

```

Data Output Messages Notifications

UPDATE 2

Query returned successfully in 77 msec.

	seat_number [PK] integer	car_number_in_train [PK] integer	train_number [PK] integer	seat_status character varying (20)	seat_price integer
53	5	1	7	Куплено	4000
54	5	1	7	Куплено	4000
55	4	1	7	Куплено	4000
56	3	1	7	Куплено	4000
57	2	1	7	Куплено	4000
58	1	1	7	Куплено	4000
59	6	1	4	Свободно	4000
60	5	1	4	Свободно	4000
61	4	1	4	Свободно	4000
62	3	1	4	Куплено	4000
63	2	1	4	Куплено	4000
64	1	1	4	Куплено	4000
65	1	1	8	Куплено	99999
66	1	2	8	Куплено	99999
67	2	2	8	Свободно	99999
68	2	1	8	Свободно	99999

Query

Query History

1

SELECT * FROM passenger_scheme.seat

2

ORDER BY seat_price

Data Output

Messages

Notifications

	seat_number [PK] integer	car_number_in_train [PK] integer	train_number [PK] integer	seat_status character varying (20)	seat_price integer
56	3	1	7	Куплено	4000
57	2	1	7	Куплено	4000
58	1	1	7	Куплено	4000
59	6	1	4	Свободно	4000
60	5	1	4	Свободно	4000
61	4	1	4	Свободно	4000
62	3	1	4	Куплено	4000
63	2	1	4	Куплено	4000
64	1	1	4	Куплено	4000
65	1	1	8	Куплено	99999
66	1	2	8	Куплено	99999
67	2	2	8	Куплено	99999
68	2	1	8	Куплено	99999

Активация Windows

Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 68 of 68

Query complete 00:00:00.084

Ln 2, Col 20

Представления

Представление 1

Query

Query History

```
1 CREATE OR REPLACE VIEW passenger_scheme.availableSeatsInfo AS
2 SELECT t.flight_number,
3        t.train_number,
4        t.train_name,
5        cit.carriage_number_in_train,
6        s.seat_number
7 FROM passenger_scheme.train t
8 JOIN passenger_scheme.carriage_in_train cit ON t.train_number = cit.train_number
9 JOIN passenger_scheme.seat s ON s.carriage_in_train_id = cit.carriage_in_train_id
10 WHERE s.seat_status = 'Свободно'
11 ORDER BY t.flight_number,
12          t.train_number,
13          t.train_name,
14          cit.carriage_number_in_train,
15          s.seat_number
```

Messages

Notifications

Data Output

CREATE VIEW

Query returned successfully in 74 msec.

Активация Windows

Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 0 of 0

Query complete 00:00:00.074

Ln 1, Col 19

passenger_scheme.availableseatsinfo/passenger/postgres@Po...					
No limit					
Query Query History					
1 SELECT * FROM passenger_scheme.availableseatsinfo					
Messages Notifications Data Output					
	flight_number	train_number	train_name	carriage_number_in_train	seat_number
	character varying (20)	integer	character varying (20)	integer	integer
1	flt_1	4	Поезд_01	1	4
2	flt_1	4	Поезд_01	1	5
3	flt_1	4	Поезд_01	1	6
4	flt_1	4	Поезд_01	2	1
5	flt_1	4	Поезд_01	2	2
6	flt_1	4	Поезд_01	3	2
7	flt_1	4	Поезд_01	3	4
8	flt_2	5	Поезд_02	1	2
9	flt_2	5	Поезд_02	1	3
10	flt_2	5	Поезд_02	1	4
11	flt_2	5	Поезд_02	1	5
12	flt_2	5	Поезд_02	1	6
13	flt_2	5	Поезд_02	2	2
Total rows: 27 of 27 Query complete 00:00:00.187 Ln 1, Col 1					

Представление 2

Query Query History					
1 CREATE OR REPLACE VIEW passenger_scheme.unsoldSeatsInfo AS					
2 SELECT t.train_number,					
3 cit.carriage_number_in_train,					
4 c.carriage_type,					
5 COUNT(s.seat_number) AS total_seat					
6 FROM passenger_scheme.train t					
7 JOIN passenger_scheme.carriage_in_train cit ON t.train_number = cit.train_number					
8 JOIN passenger_scheme.seat s ON s.carriage_in_train_id = cit.carriage_in_train_id					
9 JOIN passenger_scheme.carriage c ON c.serial_number = cit.serial_number					
10 WHERE s.seat_status = 'Свободно' AND t.departure_time BETWEEN '2024-01-17 00:00:00' AND '2024-01-18 00:00:00'					
11 GROUP BY t.train_number,					
12 cit.carriage_number_in_train,					
13 c.carriage_type					
Messages Notifications Data Output					
CREATE VIEW					
Query returned successfully in 64 msec.					
Total rows: 0 of 0 Query complete 00:00:00.064 Ln 11, Col 6					

Query Query History

```

1 SELECT * FROM passenger_scheme.unsoldseatsinfo
2

```

Messages Notifications Data Output

	train_number integer	carriage_number_in_train integer	carriage_type character varying (20)	total_seat bigint
1	4	1	Kyne	3
2	4	2	CB	2
3	4	3	CB	2
4	5	1	Kyne	5
5	5	2	Kyne	5
6	5	3	CB	3

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 6 of 6 Query complete 00:00:00.152 Ln 1, Col 1

Запросы с индексами и без

Запрос без индекса

passenger/postgres@PostgreSQL 16

Query Query History

```

1 SELECT s.seat_id, s.seat_number, cit.carriage_number_in_train, t.train_number, sch.flight_number
2 FROM passenger_scheme.seat s, passenger_scheme.carriage_in_train cit, passenger_scheme.carriage c,
3 passenger_scheme.train t, passenger_scheme.schedule sch
4 WHERE s.carriage_in_train_id = cit.carriage_in_train_id AND cit.train_number = t.train_number
5 AND t.flight_number = sch.flight_number AND cit.serial_number = c.serial_number AND c.carriage_type = 'Kyne'
6 AND s.seat_status = 'Свободно' AND sch.arrival_point = 'Москва'
7 AND sch.departure_time BETWEEN '2024-01-17 00:00:00' AND '2024-01-18 00:00:00'

```

Messages Notifications Data Output

	seat_id bigint	seat_number integer	carriage_number_in_train integer	train_number integer	flight_number character varying (10)
2	30	3	1	5	flt_2
3	42	4	1	5	flt_2
4	57	5	2	5	flt_2
5	54	5	1	5	flt_2
6	21	2	2	5	flt_2
7	34	3	2	5	flt_2
8	46	4	2	5	flt_2
9	60	6	1	5	flt_2
10	63	6	2	5	flt_2

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Total rows: 10 of 10 Query complete 00:00:00.097 Ln 7, Col 1

Запрос с индексами

Query Query History

```

1 CREATE INDEX idx_seat_id_and_number ON passenger_scheme.seat(seat_id, seat_number);
2 CREATE INDEX idx_car_in_train_id ON passenger_scheme.carriage_in_train(carriage_in_train_id);
3 CREATE INDEX idx_train_number ON passenger_scheme.train(train_number);

```

Messages Notifications Data Output

CREATE INDEX

Query returned successfully in 38 msec.

Удаление индексов

The screenshot shows the Query Tool interface for a PostgreSQL database. The title bar indicates the connection is 'passenger/postgres@PostgreSQL 16'. The interface includes a toolbar with icons for file operations, query execution, and settings. Below the toolbar, the 'Query' tab is active, displaying a list of three SQL queries:

- 1 **DROP INDEX** passenger_scheme.idx_seat_id_and_number;
- 2 **DROP INDEX** passenger_scheme.idx_car_in_train_id;
- 3 **DROP INDEX** passenger_scheme.idx_train_number;

The 'Messages' tab is also visible, showing the output of the query: 'DROP INDEX' and 'Query returned successfully in 87 msec.'

Вывод: В ходе лабораторной работы была освоена работа с Query Tool, а именно с выборкой данных, добавлением, обновлением и их удалением. Также получены навыки создания представления данных и индексирования полей, в следствии чего были проведены оптимизационные эксперименты.