

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

Отчет

по лабораторной работе №4

«Создание БД PostgreSQL в pgAdmin. Резервное копирование и восстановление БД»

по дисциплине **«Проектирование и реализация баз данных»**

Вариант 17

Автор: Аплеев Д. А.

Факультет: ПИН

Группа: К3239 Преподаватель:

Горова М. М.



Санкт-Петербург 2025

ОГЛАВЛЕНИЕ

Цель работы.....	3
Практическое задание	3
Индивидуальное задание	Ошибка! Закладка не определена.
Описание модели	Ошибка! Закладка не определена.
Выполнение	Ошибка! Закладка не определена.
Задание 1. Создать запросы	4
Задание 2. Создать представления.....	5
Задание 3. Запросы на модификацию	5
Задание 3. Индексы	9
Вывод	10

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 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.

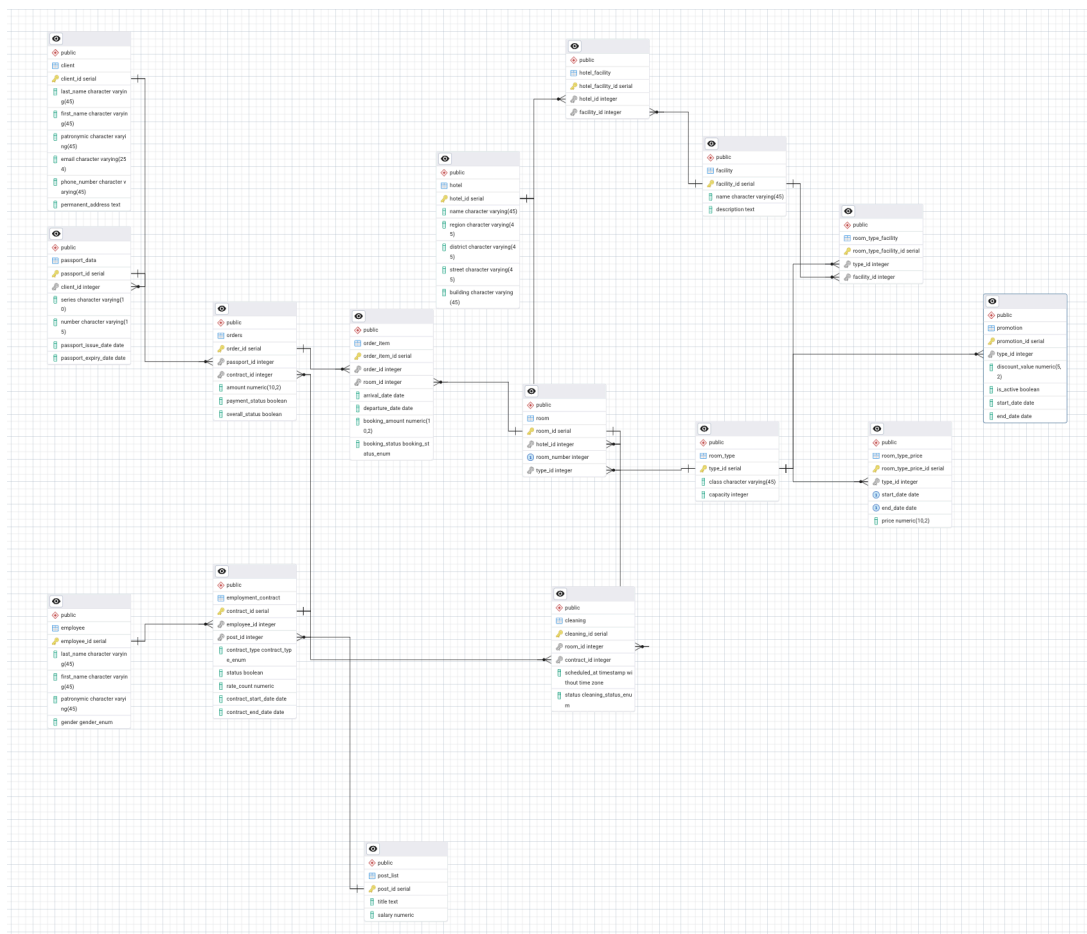


Рисунок 2. Схема логической модели базы данных

Задание 1. Создать запросы

- Вывести все личные данные клиента.

Запрос История запросов

```
1 SELECT
2 client.client_id AS "ID клиента",
3 client.last_name AS "Фамилия",
4 client.first_name AS "Имя",
5 passport_data.series AS "Серия паспорта",
6 passport_data.number AS "Номер паспорта"
7 FROM
8 client
9 JOIN
10 passport_data ON client.client_id = passport_data.client_id
11 WHERE
12 current_date <= passport_expiry_date;
```

Data Output Сообщения Notifications

	ID клиента integer	Фамилия character varying (45)	Имя character varying (45)	Серия паспорта character varying (10)	Номер паспорта character varying (15)
1	1	Sokolov	Egor	9580	440518
2	2	Ivanova	Maria	AA	1234567
3	3	Taylor	Sam	UK	998877
4	4	Schmidt	Anna	DE	C0123456
5	5	Kim	Min-soo	KR	M12345678
6	6	Garcia	Luis	ES	BA765432
7	7	Dubois	Claire	FR	12AB34567
8	8	Johnson	Emily	USA	600123456
9	9	Lopez	Carlos	MX	A12345678
10	10	Nakamura	Yuki	JP	TK1234567
11	11	Kowalski	Piotr	PL	AH1234567
12	12	Ostergard	Lars	DK	XK123456

- Найти среднее кол-во дней проживания в номерах по типам.

Запрос История запросов

```
1 SELECT
2 rt.class AS "Тип номера",
3 AVG(oi.departure_date - oi.arrival_date) AS "ср кол-во дней"
4 FROM
5 order_item oi
6 JOIN room r ON oi.room_id = r.room_id
7 JOIN room_type rt ON r.type_id = rt.type_id
8 GROUP BY
9 rt.class
10 ORDER BY
11 "ср кол-во дней" DESC;
```

Data Output Сообщения Notifications

Тип номера character varying (45)	ср кол-во дней numeric
1 Twin	7.1428571428571429
2 Double	7.0000000000000000
3 Deluxe	5.6000000000000000
4 Suite	5.4285714285714286
5 Single	4.8333333333333333

Задание 2. Создать представления

- Содержащее сведения обо всех гостях и их заказах.

```
CREATE OR REPLACE VIEW view_client_orders AS
SELECT
  c.client_id AS "ID клиента",
  c.last_name AS "Фамилия",
  c.first_name AS "Имя",
  p.series AS "серия паспорта",
  p.number AS "номер паспорта",
  COUNT(o.order_id) AS "Кол",
  COALESCE(SUM(o.amount),0) AS "Общая сумма заказов"
FROM
  client c
JOIN
  passport_data p USING (client_id)
JOIN
  orders o USING (passport_id)
GROUP BY
  c.client_id, c.last_name, c.first_name, p.series, p.number;
```

Запрос

История запросов

1

SELECT * from view_client_orders;

Data Output

Сообщения

Notifications

SQL

Showing rows: 1 to 12

	ID клиента integer	Фамилия character varying (45)	Имя character varying (45)	серия паспорта character varying (10)	номер паспорта character varying (15)	Кол bigint	Общая сумма заказов numeric
1	12	Ostergard	Lars	DK	XK123456	1	52501.00
2	1	Sokolov	Egor	9580	440518	3	99800.00
3	6	Garcia	Luis	ES	BA765432	1	30000.00
4	3	Taylor	Sam	UK	998877	2	119200.00
5	8	Johnson	Emily	USA	600123456	1	72000.00
6	5	Kim	Min-soo	KR	M12345678	1	15801.00
7	9	Lopez	Carlos	MX	A12345678	1	64000.00
8	2	Ivanova	Maria	AA	1234567	2	70000.00
9	10	Nakamura	Yuki	JP	TK1234567	1	28000.00
10	7	Dubois	Claire	FR	12AB34567	1	18501.00
11	11	Kowalski	Piotr	PL	AH1234567	1	42000.00
12	4	Schmidt	Anna	DE	C0123456	1	53000.00

Задание 3. Запросы на модификацию

Добавить нового заказа.

Запрос		История запросов	
1	▼	INSERT INTO orders (passport_id, contract_id, amount, payment_status, overall_status)	
2		VALUES (
3		(SELECT pd.passport_id	
4		FROM passport_data pd	
5		JOIN client c USING (client_id)	
6		WHERE c.email = 'egor.sokolov@yandex.ru'),	
7		(SELECT ec.contract_id	
8		FROM employment_contract ec	
9		JOIN employee e USING (employee_id)	
10		WHERE ec.status = TRUE AND e.last_name = 'Ivanov'),	
11		19000, TRUE, TRUE);	
12			



Результат:

15	15	3	5	94000.00	true	true
16	16	1	1	18000.00	true	true
17	17	1	1	19000.00	true	true

Обновление размера скидки у самого дорогого типа номера:

ЗапросИстория запросов

```
1 UPDATE promotion p
2 SET discount_value = discount_value + 50
3 FROM (
4   SELECT type_id
5   FROM room_type_price
6   GROUP BY type_id
7   ORDER BY MAX(price) DESC
8   LIMIT 1
9 ) AS t
10 WHERE p.type_id = t.type_id;
```

Результат:

	promotion_id [PK] integer	type_id integer	discount_value numeric (5,2)	is_active boolean	start_date date	end_date date
1	1	2	10.00	true	2025-06-01	2025-08-31
2	2	4	15.00	true	2025-12-15	2026-01-15
3	3	1	5.00	false	2025-03-01	2025-05-31
4	4	5	70.00	true	2025-09-01	2025-11-30
5	5	3	12.50	true	2026-04-01	2026-05-15

Этот запрос синхронизирует поле amount в таблице orders с реальными суммами по всем позициям из order_item.

type: history suppressed

```
UPDATE orders o
SET amount = sq.total_amount
FROM (
SELECT order_id, SUM(booking_amount) AS total_amount
FROM order_item
GROUP BY order_id
) AS sq
WHERE o.order_id = sq.order_id;
```

Data Output Сообщения Notifications

	order_id [PK] integer	passport_id integer	contract_id integer	amount numeric (10,2)	payment_status boolean	overall_status boolean
1	1	1	1	42000.00	true	true
2	2	2	2	8800.00	true	true
3	3	3	3	25200.00	true	false
4	4	4	4	53000.00	false	false
5	5	5	5	15801.00	true	true
6	6	6	6	30000.00	false	false
7	7	7	7	18501.00	true	false
8	8	8	8	72000.00	true	true
9	9	9	9	64000.00	false	true
10	10	10	10	28000.00	true	false
11	11	11	1	42000.00	true	true
12	12	12	2	52501.00	true	true

Удалить записи о звонках, стоимость которых наименьшая из всех звонков.

```
DELETE FROM orders o
WHERE NOT EXISTS (
    SELECT * FROM order_item oi
    WHERE oi.order_id = o.order_id);
```

До:

	order_id [PK] integer	passport_id integer	contract_id integer	amount numeric (10,2)	payment_status boolean	overall_status boolean
3	3	3	3	25200.00	true	false
4	4	4	4	53000.00	false	false
5	5	5	5	15801.00	true	true
6	6	6	6	30000.00	false	false
7	7	7	7	18501.00	true	false
8	8	8	8	72000.00	true	true
9	9	9	9	64000.00	false	true
10	10	10	10	28000.00	true	false
11	11	11	1	42000.00	true	true
12	12	12	2	52501.00	true	true
13	13	1	3	39800.00	true	true
14	14	2	4	61200.00	false	false
15	15	3	5	94000.00	true	true
16	20	1	1	18000.00	true	true

После:

	order_id [PK] integer	passport_id integer	contract_id integer	amount numeric (10,2)	payment_status boolean	overall_status boolean
2	2	2	2	8800.00	true	true
3	3	3	3	25200.00	true	false
4	4	4	4	53000.00	false	false
5	5	5	5	15801.00	true	true
6	6	6	6	30000.00	false	false
7	7	7	7	18501.00	true	false
8	8	8	8	72000.00	true	true
9	9	9	9	64000.00	false	true
10	10	10	10	28000.00	true	false
11	11	11	1	42000.00	true	true
12	12	12	2	52501.00	true	true
13	13	1	3	39800.00	true	true
14	14	2	4	61200.00	false	false
15	15	3	5	94000.00	true	true

Задание 3. Индексы

Запрос до индексов:

Запрос

История запросов

1

EXPLAIN (ANALYZE, BUFFERS)

2

SELECT *

3

FROM order_item

4

WHERE room_id = 5;

Data Output

Сообщения

Notifications

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+</

Создание индексов:

Copy

Copy to Query Editor

```
CREATE INDEX ix_order_item_room_id
ON order_item(room_id);
```

Сообщения

Запрос после индексов:

Запрос

История запросов

1

2

3

4

EXPLAIN (ANALYZE, BUFFERS)

SELECT *

FROM order_item

WHERE room_id = 5;

Data Output

Сообщения

Notifications

SQL

QUERY PLAN

text

1 Seq Scan on order_item (cost=0.00..1.38 rows=1 width=40) (actual time=0.013..0.016 rows=2 loops=...

2 Filter: (room_id = 5)

3 Rows Removed by Filter: 28

4 Buffers: shared hit=1

5 Planning Time: 0.327 ms

6 Execution Time: 0.029 ms

Вывод

В процессе выполнения работы были отработаны методы запроса данных из реляционной базы данных, включая использование разных JOIN для получения информации из нескольких связанных таблиц и агрегатных функций для выполнения расчетов. Были успешно созданы запросы на модификацию данных с использованием вложенных подзапросов.

Также были созданы представления, включая динамические представления, позволяющие получать актуальные данные за определенные периоды времени без необходимости каждый раз писать полный текст запроса. Были созданы простые индексы на полях, используемых в условиях фильтрации и соединения. С помощью EXPLAIN ANALYZE были получены и изучены планы выполнения запросов. В конце работы созданные для эксперимента индексы были удалены.

Таким образом, в ходе работы были закреплены навыки написания сложных SQL-запросов, работы с представлениями и оптимизации запросов с помощью индексов.