

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИТМО

ОТЧЕТ

по Лабораторной работе 4

по дисциплине "Проектирование и реализация баз данных"

Обучающийся: Иванов Виктор Сергеевич

Факультет прикладной информатики

Группа К3240

Направление подготовки 09.03.03 Прикладная информатика

Образовательная программа: Мобильные и сетевые технологии 2023

Преподаватель: Говорова Марина Михайловна

Санкт-Петербург  
2024/2025

## Цель работы

Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и работы с индексами.

## Практическое задание

1. Создать запросы и представления на выборку данных к базе данных.
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 и сравнить время выполнения запросов без индексов и с индексами.

## Модель базы данных

(Модель из ЛР3, таблицы: client, contract, detail, contractDetail, drawing, detailDrawing, machine, steelSheet, manufacturingStage, pricePeriod)

## ERD-схема БД

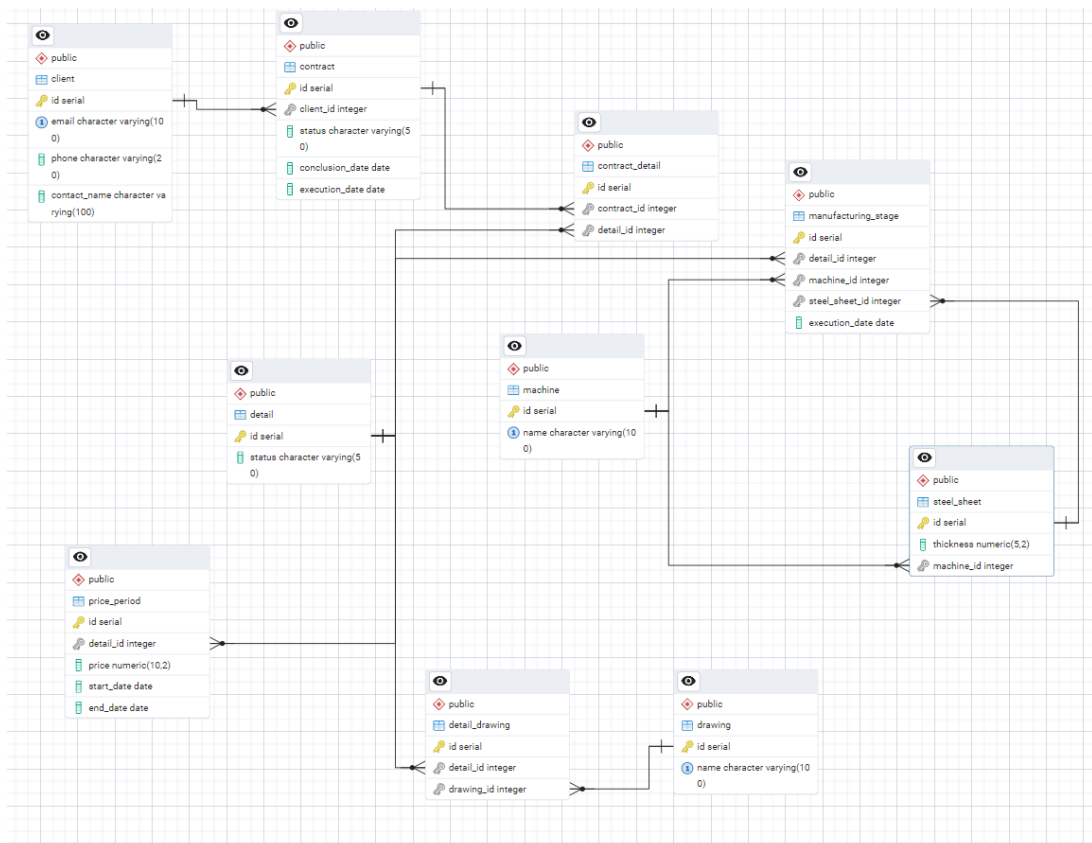


Рис. 1: ERD-схема БД

# Выполнение работы

## 1. Запросы на выборку данных

Скрипт выборки всех клиентов, у которых договоры в статусе "в работе":

```
SELECT
    c.contact_name AS "Клиент",
    ct.id AS "Номер договора",
    d.id AS "Номер детали",
    d.status AS "Статус детали"
FROM client c
JOIN contract ct ON c.id = ct.client_id
JOIN contract_detail cd ON ct.id = cd.contract_id
JOIN detail d ON d.id = cd.detail_id
WHERE ct.status = 'в работе'
ORDER BY c.contact_name, ct.id, d.id;
```

Листинг 1: Запрос для получения списка клиентов

	Клиент character varying (100)	Номер договора integer	Номер детали integer	Статус детали character varying (50)
1	Белова Н.М.	11	11	выполнена
2	Демидов И.Г.	15	15	проект
3	Жуков Ф.А.	13	13	в производстве
4	Иванов И.И.	1	1	проект
5	Иванов И.И.	1	2	в производстве
6	Иванова Л.С.	5	5	выполнена
7	Иванова Л.С.	5	6	проект
8	Кузнецов В.Н.	7	7	в производстве
9	Миронова Е.Д.	9	9	проект
10	Рогов П.П.	4	4	в производстве
11	Рогов П.П.	4	5	выполнена

Рис. 2: Результат выполнения запроса на выборку

Скрипт выборки деталей, которые изготавливались на станках, и указания даты изготовления вместе с названием станка:

```
SELECT
    d.id AS "Номер детали",
    d.status AS "Статус детали",
    m.name AS "Название станка",
    ms.execution_date AS "Дата изготовления"
FROM detail d
JOIN manufacturing_stage ms ON d.id = ms.detail_id
JOIN machine m ON m.id = ms.machine_id
ORDER BY ms.execution_date DESC;
```

Листинг 2: Запрос для получения списка деталей

	Номер детали integer	Статус детали character varying (50)	Название станка character varying (100)	Дата изготовления date
1	6	проект	Станок 04	2025-04-15
2	5	выполнена	Станок 03	2025-04-14
3	4	в производстве	Станок 15	2025-04-13
4	15	проект	Станок 14	2025-04-12
5	14	выполнена	Станок 13	2025-04-11
6	13	в производстве	Станок 12	2025-04-10
7	12	проект	Станок 11	2025-04-09
8	11	выполнена	Станок 10	2025-04-08
9	10	в производстве	Станок 09	2025-04-07
10	9	проект	Станок 08	2025-04-06
11	8	выполнена	Станок 07	2025-04-05
12	7	в производстве	Станок 06	2025-04-04
13	6	проект	Станок 05	2025-04-03
14	5	выполнена	Станок 04	2025-04-02
15	4	в производстве	Станок 03	2025-04-01
16	2	в производстве	Станок 02	2025-03-21
17	1	проект	Станок 01	2025-03-20

Рис. 3: Результат выполнения запроса на выборку

## 2. Представления (Views)

Создадим представление active contracts, которое формирует таблицу, содержащую информацию о действующих договорах в базе данных.

```
CREATE VIEW active_contracts AS
SELECT ct.id, c.contact_name, ct.conclusion_date
FROM contract ct
JOIN client c ON c.id = ct.client_id
WHERE ct.status = 'в работе';
```

Листинг 3: Создание active contracts

Далее через *SELECT \* FROM active contracts* выводим результат - само представление

	id integer	contact_name character varying (100)	conclusion_date date
1	1	Иванов И.И.	2025-03-01
2	17	Петрова А.А.	2025-03-12
3	4	Рогов П.П.	2025-03-10
4	5	Иванова Л.С.	2025-02-15
5	7	Кузнецов В.Н.	2025-04-01
6	9	Миронова Е.Д.	2025-01-25
7	11	Белова Н.М.	2025-03-05
8	13	Жуков Ф.А.	2025-03-15
9	15	Демидов И.Г.	2025-04-10

Рис. 4: Создание представления

### 3. Модификация данных (INSERT, UPDATE, DELETE с подзапросами)

#### INSERT с подзапросом

Перед выполнением запроса выведем таблицу contract.

	id [PK] integer	client_id integer	status character varying (50)	conclusion_date date	execution_date date
1	1	1	в работе	2025-03-01	[null]
2	2	2	завершен	2024-11-10	2025-01-20
3	3	3	отменен	2025-01-05	[null]
4	4	4	в работе	2025-03-10	[null]
5	5	5	в работе	2025-02-15	[null]
6	6	6	завершен	2024-12-01	2025-01-30
7	7	7	в работе	2025-04-01	[null]
8	8	8	отменен	2025-02-20	[null]
9	9	9	в работе	2025-01-25	[null]
10	10	10	завершен	2024-10-10	2024-12-15
11	11	11	в работе	2025-03-05	[null]
12	12	12	завершен	2025-01-01	2025-02-25
13	13	13	в работе	2025-03-15	[null]
14	14	14	отменен	2025-02-01	[null]
15	15	15	в работе	2025-04-10	[null]
16	16	1	завершен	2024-09-05	2024-11-01
17	17	2	в работе	2025-03-12	[null]
18	18	3	отменен	2025-02-15	[null]

Рис. 5: Состояние таблицы contract до выполнения INSERT

```
INSERT INTO contract (client_id, status, conclusion_date)
SELECT id, 'в работе', CURRENT_DATE
FROM client
WHERE contact_name = 'Петрова А.А.';
```

Листинг 4: Добавление нового договора для клиента через подзапрос

	id [PK] integer	client_id integer	status character varying (50)	conclusion_date date	execution_date date
1	1	1	в работе	2025-03-01	[null]
2	2	2	завершен	2024-11-10	2025-01-20
3	3	3	отменен	2025-01-05	[null]
4	4	4	в работе	2025-03-10	[null]
5	5	5	в работе	2025-02-15	[null]
6	6	6	завершен	2024-12-01	2025-01-30
7	7	7	в работе	2025-04-01	[null]
8	8	8	отменен	2025-02-20	[null]
9	9	9	в работе	2025-01-25	[null]
10	10	10	завершен	2024-10-10	2024-12-15
11	11	11	в работе	2025-03-05	[null]
12	12	12	завершен	2025-01-01	2025-02-25
13	13	13	в работе	2025-03-15	[null]
14	14	14	отменен	2025-02-01	[null]
15	15	15	в работе	2025-04-10	[null]
16	16	1	завершен	2024-09-05	2024-11-01
17	17	2	в работе	2025-03-12	[null]
18	18	3	отменен	2025-02-15	[null]
19	19	2	в работе	2025-04-28	[null]

Рис. 6: Состояние таблицы contract после выполнения INSERT

#### UPDATE с подзапросом

Перед выполнением запроса проверим статусы деталей.

	id [PK] integer	status character varying (50)
1	1	проект
2	2	в производстве
3	3	выполнена
4	4	в производстве
5	5	выполнена
6	6	проект
7	7	в производстве
8	8	выполнена
9	9	проект
10	10	в производстве
11	11	выполнена
12	12	проект
13	13	в производстве
14	14	выполнена
15	15	проект
16	16	в производстве
17	17	выполнена
18	18	проект

Рис. 7: Статусы деталей до выполнения UPDATE

```
UPDATE detail
SET status = 'выполнена'
WHERE id IN (
    SELECT cd.detail_id
    FROM contract_detail cd
    JOIN contract ct ON cd.contract_id = ct.id
    WHERE ct.status = 'завершен'
);
```

Листинг 5: Обновление статуса деталей через подзапрос

## DELETE с подзапросом

Перед выполнением запроса проверим наличие этапов изготовления для деталей со статусом «проект».

```
DELETE FROM manufacturing_stage
WHERE detail_id IN (
    SELECT id
    FROM detail
    WHERE status = 'проект'
);
```

	id [PK] integer	status character varying (50)
1	1	проект
2	2	в производстве
3	4	в производстве
4	5	выполнена
5	8	выполнена
6	9	проект
7	11	выполнена
8	13	в производстве
9	14	выполнена
10	15	проект
11	16	в производстве
12	17	выполнена
13	18	проект
14	7	выполнена
15	10	выполнена
16	6	выполнена
17	12	выполнена
18	3	выполнена

Рис. 8: Статусы деталей после выполнения UPDATE

	id [PK] integer	detail_id integer	machine_id integer	steel_sheet_id integer	execution_date date
1	1	1	1	1	2025-03-20
2	2	2	2	2	2025-03-21
3	3	4	3	3	2025-04-01
4	4	5	4	4	2025-04-02
5	5	6	5	5	2025-04-03
6	6	7	6	6	2025-04-04
7	7	8	7	7	2025-04-05
8	8	9	8	8	2025-04-06
9	9	10	9	9	2025-04-07
10	10	11	10	10	2025-04-08
11	11	12	11	11	2025-04-09
12	12	13	12	12	2025-04-10
13	13	14	13	13	2025-04-11
14	14	15	14	14	2025-04-12
15	15	4	15	15	2025-04-13
16	16	5	3	3	2025-04-14
17	17	6	4	4	2025-04-15

Рис. 9: Этапы изготовления до выполнения DELETE

---

Листинг 6: Удаление этапов изготовления для проектных деталей через подзапрос

	id [PK] integer	detail_id integer	machine_id integer	steel_sheet_id integer	execution_date date
1	2	2	2	2	2025-03-21
2	3	4	3	3	2025-04-01
3	4	5	4	4	2025-04-02
4	5	6	5	5	2025-04-03
5	6	7	6	6	2025-04-04
6	7	8	7	7	2025-04-05
7	9	10	9	9	2025-04-07
8	10	11	10	10	2025-04-08
9	11	12	11	11	2025-04-09
10	12	13	12	12	2025-04-10
11	13	14	13	13	2025-04-11
12	15	4	15	15	2025-04-13
13	16	5	3	3	2025-04-14
14	17	6	4	4	2025-04-15

Рис. 10: Этапы изготовления после выполнения DELETE

## 4. Создание индексов и анализ планов выполнения

### Запрос без индекса

Выполняем запрос поиска договоров со статусом «в работе» без использования индекса.

```
EXPLAIN ANALYZE
SELECT *
FROM contract
WHERE status = 'в работе';
```

Листинг 7: Запрос без индекса

	QUERY PLAN text
1	Seq Scan on contract (cost=0.00..2.49 rows=43 width=32) (actual time=0.015..0.044 rows=43 loops=...
2	Filter: ((status)::text = 'в работе'::text)
3	Rows Removed by Filter: 76
4	Planning Time: 0.088 ms
5	Execution Time: 0.057 ms

Рис. 11: План выполнения запроса без индекса

### Создание индекса

Создаем индекс на столбец status в таблице contract.

```
CREATE INDEX idx_contract_status ON contract(status);
```

Листинг 8: Создание индекса на contract(status)

### Запрос с индексом

Выполняем тот же запрос после создания индекса.

```
EXPLAIN ANALYZE
SELECT *
```



```
FROM contract
WHERE status = 'в работе';
```

Листинг 9: Запрос с индексом

	QUERY PLAN
	text
1	Seq Scan on contract (cost=0.00..2.49 rows=43 width=32) (actual time=0.010..0.018 rows=43 loops=...
2	Filter: ((status)::text = 'в работе'::text)
3	Rows Removed by Filter: 76
4	Planning Time: 0.821 ms
5	Execution Time: 0.026 ms

Рис. 12: План выполнения запроса с индексом

## Удаление индекса

После проверки удаляем индекс.

```
DROP INDEX idx_contract_status;
```

Листинг 10: Удаление индекса

## Сравнение времени выполнения

По результатам EXPLAIN видно, что после создания индекса время выполнения запроса уменьшилось. - Без индекса:  $[0.057\ ms]$  - С индексом:  $[0.026\ ms]$

Таким образом, использование индексов позволяет значительно ускорить выполнение запросов, особенно при фильтрации по часто используемым столбцам.

## Выводы

В ходе лабораторной работы были разработаны запросы для выборки и модификации данных, созданы представления, проанализированы планы выполнения запросов с использованием индексов. Была подтверждена эффективность применения индексов для ускорения выполнения запросов.