**Министерство науки и высшего образования Российской Федерации**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Факультет безопасности информационных технологий**

**Направление подготовки: 11.03.03**

**Образовательная программа: Безопасность информационных технологий**

**Дисциплина:**

«Информационная безопасность баз данных»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

« Реализация БД в рамках СУБД »

**Выполнил:**

*A blue logo with a white background

Description automatically generated*Рядовой Т.С., студент группы N3352, поток ИББД.Nб3 1.5

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

**Проверил:**

Салихов Максим Русланович

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(отметка о выполнении)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(дата)

Содержание

[Введение 3](#_Toc192030410)

[1 Ход работы 4](#_Toc192030411)

[1.1 Выбор СУБД 4](#_Toc192030412)

[1.2 Создание БД и заполнение данными 5](#_Toc192030413)

[1.3 Индексация таблиц 6](#_Toc192030414)

[1.4 Установка взаимосвязей между таблицами 6](#_Toc192030415)

[1.5 Заполнение данными 7](#_Toc192030416)

[1.6 Создание представлений 9](#_Toc192030417)

[1.6.1 Представление №1 – Текущие и новые заказы 9](#_Toc192030418)

[1.6.2 Представление №2 - Отчет по материалам 10](#_Toc192030419)

[1.6.3 Представление №3 – Задания для мастера 10](#_Toc192030420)

[1.6.4 Представление №4 – Отчет о выполненных заказах 11](#_Toc192030421)

[Заключение 12](#_Toc192030422)

Введение

Цель работы: Получение навыков по работе с современными системами управления базами данных (СУБД).

# Ход работы

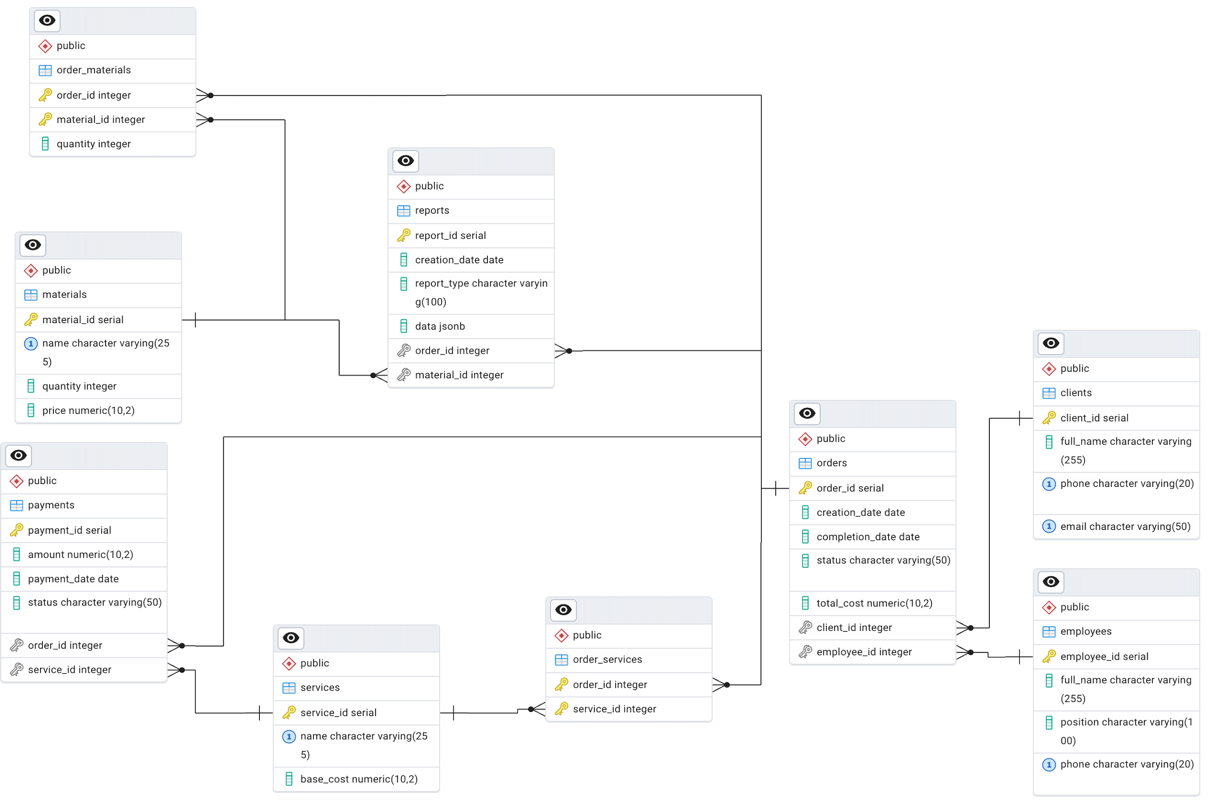
Задание:

1. Выбрать систему управления базами данных (СУБД) и обосновать свой выбор.
2. Создать БД в выбранной СУБД на основе итоговой разработанной схемы отношений из ЛР 1. Заполнить созданную БД информацией.
3. Индексировать таблицы. Добавить индексы для атрибутов, по которым происходит объединение таблиц, а также атрибуты, по которым выполняется поиск/фильтрация данных.
4. Установить взаимосвязи между таблицами.
5. Создать представления, составленные в пункте 5 лабораторной работы №1.

## Выбор СУБД

В качестве инструментария для создания БД выбрана СУБД PostgreSQL с графическим интерфейсом pgAdmin4.  
Обоснование выбора:

* PostgreSQL является одной из наиболее популярных и мощных СУБД с открытым исходным кодом;
* Поддержка сложных запросов, транзакций и индексов;
* Возможность работы с большими объемами данных;
* Широкая поддержка сообществом и обширная документация.



1. ER-диаграмма базы данных

## Создание БД и заполнение данными

1. Создание базы данных

CREATE DATABASE repair\_workshop

WITH

OWNER = postgres

ENCODING = 'UTF8'

TABLESPACE = pg\_default

CONNECTION LIMIT = -1

IS\_TEMPLATE = False;

1. Создание таблиц

-- Таблица "Клиенты"

CREATE TABLE IF NOT EXISTS public.clients (

client\_id SERIAL PRIMARY KEY,

full\_name VARCHAR(255) NOT NULL,

phone VARCHAR(20) UNIQUE NOT NULL,

email VARCHAR(50) UNIQUE

);

-- Таблица "Заказы"

CREATE TABLE IF NOT EXISTS public.orders (

order\_id SERIAL PRIMARY KEY,

creation\_date DATE NOT NULL,

completion\_date DATE,

status VARCHAR(50) CHECK (status IN ('Новый', 'В процессе', 'Завершен', 'Отменен')) NOT NULL,

total\_cost NUMERIC(10, 2) CHECK (total\_cost >= 0),

client\_id INTEGER REFERENCES clients(client\_id) ON DELETE CASCADE

);

-- Таблица "Сотрудники"

CREATE TABLE IF NOT EXISTS public.employees (

employee\_id SERIAL PRIMARY KEY,

full\_name VARCHAR(255) NOT NULL,

position VARCHAR(100) NOT NULL,

phone VARCHAR(20) UNIQUE NOT NULL

);

-- Таблица "Материалы"

CREATE TABLE IF NOT EXISTS public.materials (

material\_id SERIAL PRIMARY KEY,

name VARCHAR(255) UNIQUE NOT NULL,

quantity INTEGER CHECK (quantity >= 0),

price NUMERIC(10, 2) CHECK (price >= 0)

);

-- Таблица "Услуги"

CREATE TABLE IF NOT EXISTS public.services (

service\_id SERIAL PRIMARY KEY,

name VARCHAR(255) UNIQUE NOT NULL,

base\_cost NUMERIC(10, 2) CHECK (base\_cost >= 0)

);

-- Таблица "Оплаты"

CREATE TABLE IF NOT EXISTS public.payments (

payment\_id SERIAL PRIMARY KEY,

amount NUMERIC(10, 2) CHECK (amount >= 0),

payment\_date DATE NOT NULL,

status VARCHAR(50) CHECK (status IN ('Оплачено', 'Частично оплачено', 'Не оплачено')) NOT NULL,

order\_id INTEGER REFERENCES orders(order\_id) ON DELETE CASCADE

);

-- Таблица "Отчеты"

CREATE TABLE IF NOT EXISTS public.reports (

report\_id SERIAL PRIMARY KEY,

creation\_date DATE NOT NULL,

report\_type VARCHAR(100) NOT NULL,

data JSONB,

order\_id INTEGER REFERENCES orders(order\_id) ON DELETE CASCADE

);

-- Таблица "Материалы заказа"

CREATE TABLE IF NOT EXISTS public.order\_materials (

order\_id INTEGER REFERENCES public.orders(order\_id) ON DELETE CASCADE,

material\_id INTEGER REFERENCES public.materials(material\_id) ON DELETE CASCADE,

quantity INTEGER NOT NULL CHECK (quantity > 0),

PRIMARY KEY (order\_id, material\_id)

);

-- Таблица "Услуги заказа"

CREATE TABLE IF NOT EXISTS public.order\_services (

order\_id INT REFERENCES public.orders(order\_id) ON DELETE CASCADE,

service\_id INT REFERENCES public.services(service\_id) ON DELETE CASCADE,

PRIMARY KEY (order\_id, service\_id)

);

## Индексация таблиц

1. Индексация таблиц

-- Индексы для таблицы "Заказы"

CREATE INDEX idx\_orders\_client\_id ON public.orders(client\_id);

-- Индексы для таблицы "Оплаты"

CREATE INDEX idx\_payments\_order\_id ON public.payments(order\_id);

-- Индексы для таблицы "Отчеты"

CREATE INDEX idx\_reports\_order\_id ON public.reports(order\_id);

## Установка взаимосвязей между таблицами

1. Установка взаимосвязей

-- reports

ALTER TABLE public.reports

ADD COLUMN material\_id INT;

ALTER TABLE public.reports

ADD CONSTRAINT fk\_material FOREIGN KEY (material\_id)

REFERENCES public.materials(material\_id)

ON DELETE SET NULL;

-- payments

ALTER TABLE public.payments

ADD COLUMN service\_id INT;

ALTER TABLE public.payments

ADD CONSTRAINT fk\_service FOREIGN KEY (service\_id)

REFERENCES public.services(service\_id)

ON DELETE SET NULL;

-- orders

ALTER TABLE public.orders

ADD COLUMN employee\_id INT;

ALTER TABLE public.orders

ADD CONSTRAINT fk\_employee FOREIGN KEY (employee\_id)

REFERENCES public.employees(employee\_id)

ON DELETE SET NULL;

## Заполнение данными

1. Заполнение таблиц данными

-- Клиенты (10 записей)

INSERT INTO public.clients (full\_name, phone, email) VALUES

('Иванов Иван Иванович', '+79111234567', 'ivanov@mail.ru'),

('Петров Петр Петрович', '+79119876543', 'petrov@gmail.com'),

('Сидорова Анна Сергеевна', '+79112345678', 'sidorova@yandex.ru'),

('Васильев Василий Васильевич', '+79119871234', 'vasiliev@mail.ru'),

('Кузнецова Ольга Николаевна', '+79117654321', 'kuznetsova@bk.ru'),

('Смирнов Алексей Владимирович', '+79114567890', 'smirnov@outlook.com'),

('Андреев Андрей Андреевич', '+79115678901', 'andreev@gmail.com'),

('Морозова Марина Павловна', '+79113456789', 'morozova@yandex.ru'),

('Григорьев Григорий Григорьевич', '+79112349876', 'grigoriev@mail.ru'),

('Федорова Дарья Викторовна', '+79119875432', 'fedorova@bk.ru');

-- Сотрудники (10 записей)

INSERT INTO public.employees (full\_name, position, phone) VALUES

('Смирнов Алексей Владимирович', 'Мастер', '+79111234569'),

('Кузнецова Ольга Ивановна', 'Кладовщик', '+79111234570'),

('Васильев Дмитрий Сергеевич', 'Администратор', '+79111234571'),

('Морозов Павел Игоревич', 'Инженер', '+79111234572'),

('Тимофеев Артем Викторович', 'Сервисный специалист', '+79111234573'),

('Борисова Наталья Петровна', 'Бухгалтер', '+79111234574'),

('Капустин Виктор Александрович', 'Мастер', '+79111234575'),

('Семенова Екатерина Алексеевна', 'Кладовщик', '+79111234576'),

('Егоров Николай Олегович', 'Инженер', '+79111234577'),

('Фролова Ирина Владимировна', 'Оператор', '+79111234578');

-- Заказы (10 записей)

INSERT INTO public.orders (creation\_date, completion\_date, status, total\_cost, client\_id, employee\_id) VALUES

('2024-01-01', '2024-01-10', 'Завершен', 5000.00, 1, 1),

('2024-01-05', NULL, 'В процессе', 3000.00, 2, 2),

('2024-01-10', NULL, 'Новый', 2000.00, 3, 3),

('2024-02-01', NULL, 'Новый', 4000.00, 4, 4),

('2024-02-05', '2024-02-15', 'Завершен', 3500.00, 5, 5),

('2024-02-10', NULL, 'В процессе', 2500.00, 6, 6),

('2024-02-15', NULL, 'Новый', 1500.00, 7, 7),

('2024-02-20', NULL, 'Новый', 1800.00, 8, 8),

('2024-02-25', '2024-03-01', 'Завершен', 5000.00, 9, 9),

('2024-03-01', NULL, 'В процессе', 2200.00, 10, 10);

-- Материалы (10 записей)

INSERT INTO public.materials (name, quantity, price) VALUES

('Винты', 100, 10.00),

('Гайки', 200, 5.00),

('Шурупы', 150, 8.00),

('Провода', 300, 15.00),

('Платы', 50, 500.00),

('Разъемы', 120, 20.00),

('Клей', 75, 25.00),

('Термопаста', 100, 50.00),

('Чипы', 30, 700.00),

('Конденсаторы', 500, 2.00);

-- Услуги (10 записей)

INSERT INTO public.services (name, base\_cost) VALUES

('Ремонт компьютера', 1000.00),

('Замена жесткого диска', 500.00),

('Установка программного обеспечения', 300.00),

('Чистка ноутбука', 700.00),

('Диагностика', 400.00),

('Замена термопасты', 600.00),

('Настройка сети', 800.00),

('Ремонт блока питания', 1200.00),

('Замена экрана', 2500.00),

('Перепайка компонентов', 1800.00);

-- Оплаты (10 записей)

INSERT INTO public.payments (amount, payment\_date, status, order\_id, service\_id) VALUES

(5000.00, '2024-01-10', 'Оплачено', 1, 1),

(1500.00, '2024-01-06', 'Частично оплачено', 2, 2),

(3500.00, '2024-02-15', 'Оплачено', 5, 3),

(1000.00, '2024-02-16', 'Частично оплачено', 6, 4),

(1800.00, '2024-02-26', 'Оплачено', 9, 5),

(1200.00, '2024-03-01', 'Оплачено', 10, 6),

(3000.00, '2024-02-21', 'Оплачено', 8, 7),

(700.00, '2024-02-22', 'Частично оплачено', 7, 8),

(2500.00, '2024-02-28', 'Оплачено', 4, 9),

(600.00, '2024-02-27', 'Частично оплачено', 3, 10);

-- Отчеты (10 записей)

INSERT INTO public.reports (creation\_date, report\_type, data, order\_id, material\_id) VALUES

('2024-01-10', 'Финансовый отчет', '{"total\_cost": 5000, "paid": 5000}', 1, 1),

('2024-01-06', 'Отчет по материалам', '{"materials\_used": ["Винты", "Гайки"]}', 2, 2),

('2024-02-15', 'Финансовый отчет', '{"total\_cost": 3500, "paid": 3500}', 5, 3),

('2024-02-16', 'Отчет по материалам', '{"materials\_used": ["Шурупы"]}', 6, 4),

('2024-02-20', 'Финансовый отчет', '{"total\_cost": 1800, "paid": 1800}', 8, 5),

('2024-02-22', 'Отчет по материалам', '{"materials\_used": ["Провода", "Клей"]}', 7, 6),

('2024-02-25', 'Финансовый отчет', '{"total\_cost": 5000, "paid": 5000}', 9, 7),

('2024-02-27', 'Отчет по материалам', '{"materials\_used": ["Термопаста"]}', 3, 8),

('2024-02-28', 'Финансовый отчет', '{"total\_cost": 2500, "paid": 2500}', 4, 9),

('2024-03-01', 'Финансовый отчет', '{"total\_cost": 2200, "paid": 2200}', 10, 10);

-- Связь заказов и материалов

INSERT INTO public.order\_materials (order\_id, material\_id, quantity) VALUES

(1, 1, 10),

(1, 2, 20),

(2, 3, 15),

(2, 4, 10),

(3, 5, 2),

(4, 6, 5),

(5, 7, 3),

(6, 8, 4),

(7, 9, 1),

(8, 10, 50),

(9, 1, 10),

(9, 3, 12),

(10, 2, 5),

(10, 5, 3);

-- Связь заказов и услуг

INSERT INTO public.order\_services (order\_id, service\_id) VALUES

(1, 1),

(1, 2),

(2, 3),

(2, 4),

(3, 5),

(4, 6),

(5, 7),

(6, 8),

(7, 9),

(8, 10),

(9, 1),

(9, 3),

(10, 2),

(10, 4);

## Создание представлений

### Представление №1 – Текущие и новые заказы

1. Представление №1

CREATE VIEW public.current\_orders AS

SELECT

o.order\_id,

o.status,

c.full\_name AS client\_name,

o.creation\_date,

o.total\_cost

FROM public.orders o

JOIN public.clients c ON o.client\_id = c.client\_id

WHERE o.status NOT IN ('Завершен', 'Отменен'); -- only active orders

A screen shot of a computer

Description automatically generated

1. Представление №1

### Представление №2 - Отчет по материалам

1. Представление №2

CREATE VIEW public.material\_reports AS

SELECT

m.name AS material\_name,

m.material\_id AS material\_id,

m.quantity AS stock\_quantity,

m.price AS unit\_price,

COALESCE(SUM(om.quantity), 0) AS used\_in\_orders

FROM public.materials m

LEFT JOIN public.order\_materials om ON m.material\_id = om.material\_id

GROUP BY m.material\_id;

A screenshot of a computer screen

Description automatically generated

1. Представление №2

### Представление №3 – Задания для мастера

1. Представление №3

CREATE VIEW public.master\_tasks AS

SELECT

o.order\_id,

c.full\_name AS client\_name,

STRING\_AGG(DISTINCT s.name, ', ') AS service\_list,

STRING\_AGG(DISTINCT m.name, ', ') AS material\_list

FROM public.orders o

JOIN public.clients c ON o.client\_id = c.client\_id

LEFT JOIN public.order\_services os ON o.order\_id = os.order\_id

LEFT JOIN public.services s ON os.service\_id = s.service\_id

LEFT JOIN public.order\_materials om ON o.order\_id = om.order\_id

LEFT JOIN public.materials m ON om.material\_id = m.material\_id

GROUP BY o.order\_id, c.full\_name;

A computer screen shot of a service

Description automatically generated

1. Представление №3

### Представление №4 – Отчет о выполненных заказах

1. Представление №4

CREATE VIEW public.completed\_work\_reports AS

SELECT

o.order\_id,

e.full\_name AS employee\_name,

o.completion\_date,

STRING\_AGG(DISTINCT m.name, ', ') AS used\_materials

FROM public.orders o

JOIN public.employees e ON o.employee\_id = e.employee\_id

LEFT JOIN public.order\_materials om ON o.order\_id = om.order\_id

LEFT JOIN public.materials m ON om.material\_id = m.material\_id

WHERE o.status = 'Завершен'

GROUP BY o.order\_id, e.full\_name, o.completion\_date;

A screen shot of a computer

Description automatically generated

1. Представление №4

Заключение

В ходе выполнения лабораторной работы была создана база данных для ремонтной мастерской, заполнена тестовыми данными, проиндексированы ключевые атрибуты и созданы представления для удобства работы с данными.