|  |  |
| --- | --- |
|  |  |
| МИНОБРНАУКИ РОССИИ |  |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |  |

**Институт информационных технологий**

КАФЕДРА ЦИФРОВОЙ ТРАНСФОРМАЦИИ

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №3**

Проектирование концептуальной схемы данных функциональной области

**по дисциплине**

«Разработка баз данных»

Выполнил студент группы ИКБО-41-23 Трофимов А.А.

Принял преподаватель Мажей Я.В.

Практические работы работа выполнены «\_\_\_»\_\_\_\_\_\_\_2025 г.

«Зачтено» «\_\_\_»\_\_\_\_\_\_\_2025 г.

Москва 2025

**Цель работы:** Работа направлена на формирование глубокого понимания и практического применения инструментов для реализации сложной бизнес-логики непосредственно на уровне базы данных.

Таблицы базы данных:

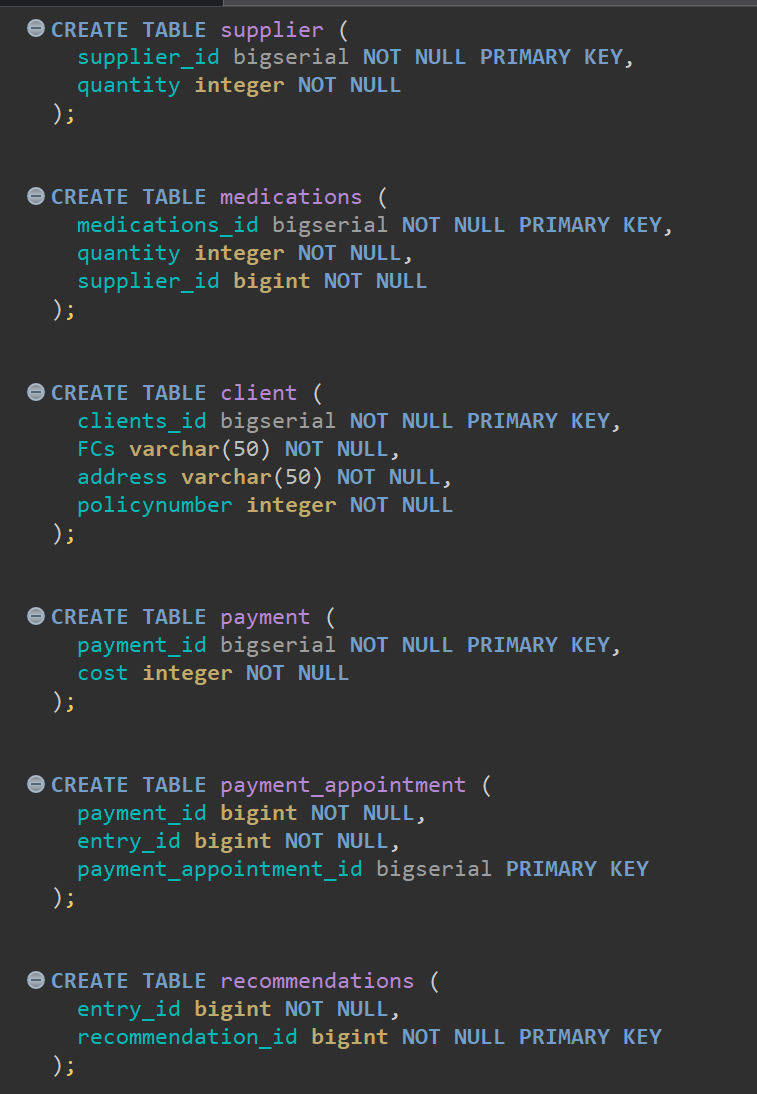


Рисунок 1

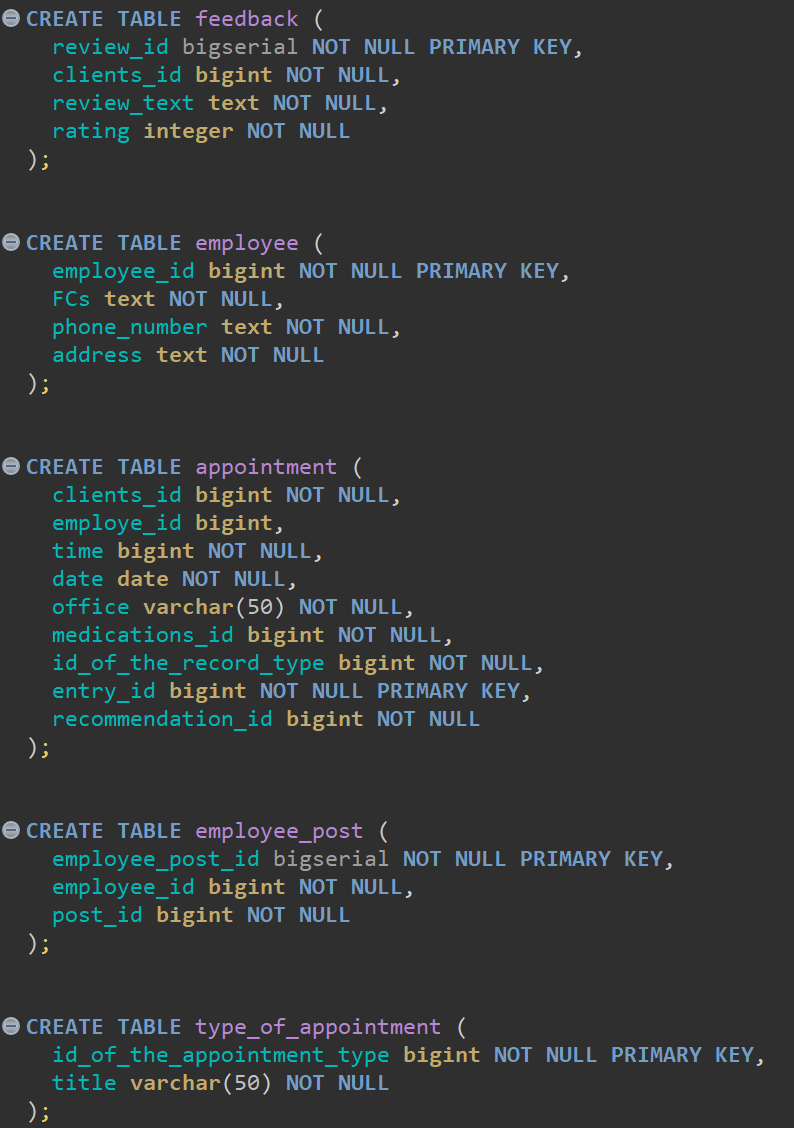


Рисунок 2

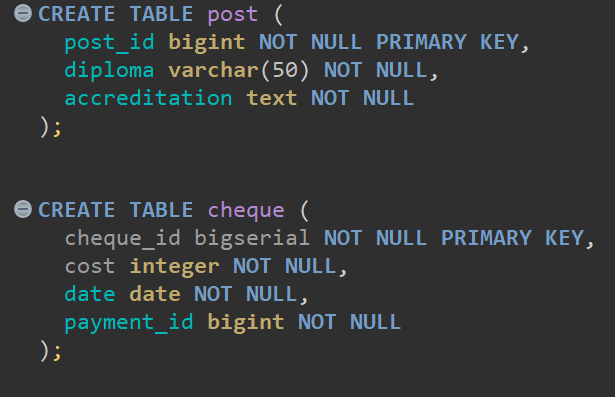


Рисунок 3

**Задание 1: использование оператора CASE**

**1.** Составить запрос, использующий поисковое выражение CASE для

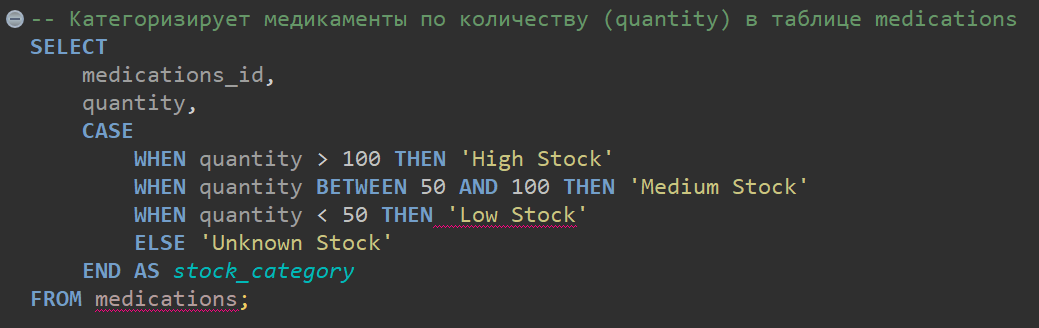
категоризации данных по какому-либо числовому признаку из вашей БД  


Рисунок 4 – запрос 1.1

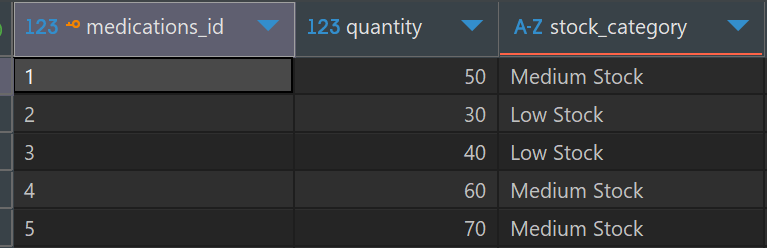


Рисунок 5 – результат запроса 1.1

2. Составить запрос, в котором оператор CASE используется внутри

агрегатной функции (например, SUM или COUNT) для выполнения

условной агрегации

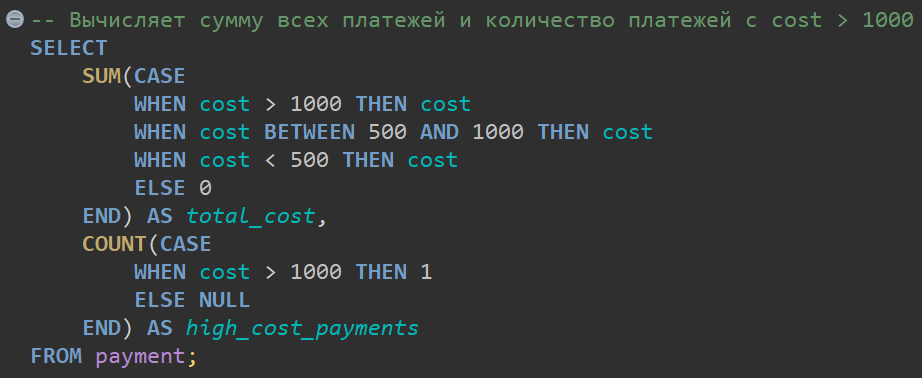


Рисунок 6 – запрос 1.2

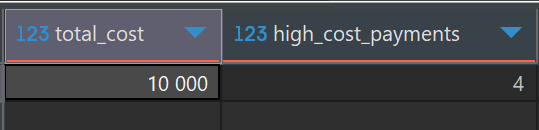


Рисунок 7 – результат запроса 1.2

**Задание 2**: использование подзапросов. Составить и выполнить тризапроса, демонстрирующих разные типы подзапросов.

1. Скалярный подзапрос: найти все записи в таблице, у которых значение в некотором числовом столбце превышает среднее (или максимальное/минимальное) значение по этому столбцу.

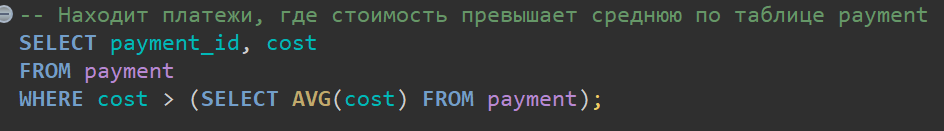


Рисунок 8 – запрос 2.1

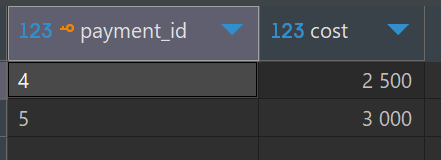


Рисунок 9 – результат запроса 2.1

2. Многострочный подзапрос с IN: вывести информацию из одной

таблицы на основе идентификаторов, полученных из связанной таблицы по определенному критерию (в данном случае, обязательно по дате).

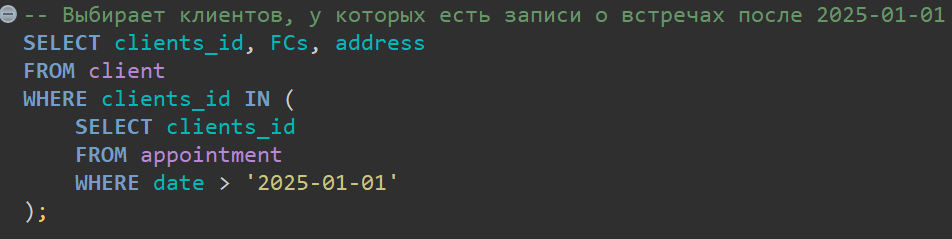


Рисунок 10 – запрос 2.2

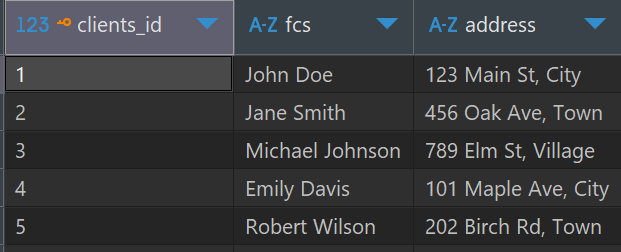


Рисунок 11 – результат запроса 2.2

3. Коррелированный подзапрос с EXISTS: найти все записи из

родительской таблицы, для которых существует хотя бы одна связанная запись в дочерней таблице, удовлетворяющая текстовому условию.

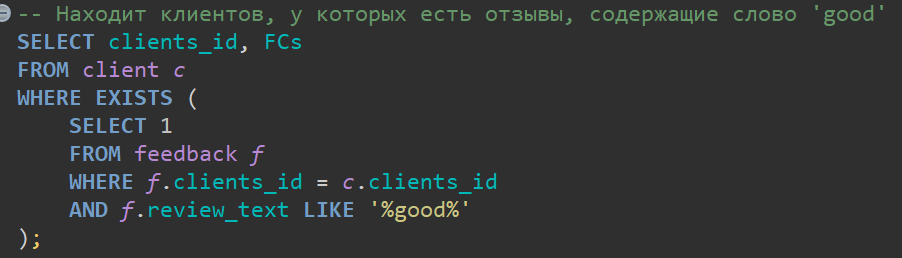


Рисунок 11 – запрос 2.3

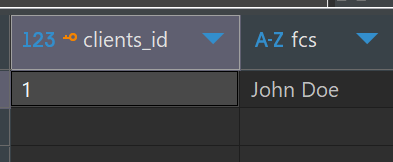


Рисунок 12 – результат запроса 2.3

4. Альтернативное решение с JOIN: решите задачу из пункта выше (2.3, Коррелированный подзапрос с EXISTS), но на этот раз с использованием оператора соединения JOIN.

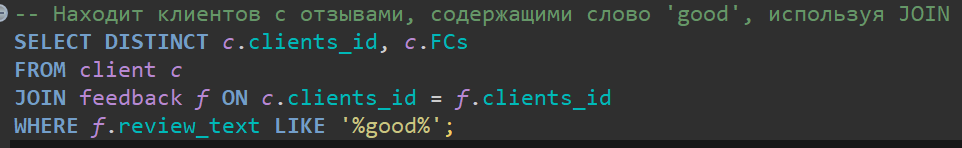


Рисунок 13 – запрос 2.4

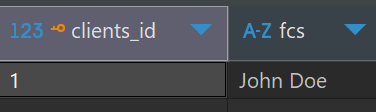


Рисунок 14 – результат запроса 2.4

**Задание 3: использование обобщенных табличных выражений (CTE).**

1. Стандартное CTE: переписать запрос из Задания 2.3 (с

коррелированным подзапросом) с использованием обобщенного

табличного выражения (CTE)

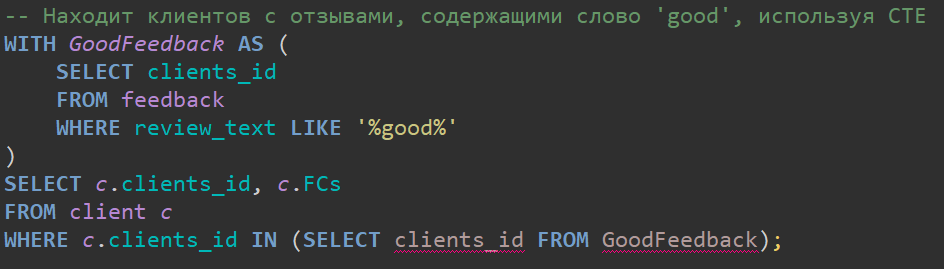


Рисунок 15 – запрос 3.1

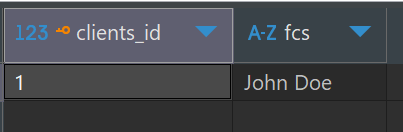


Рисунок 16 – результат запроса 3.1

2. Рекурсивное CTE: используя имеющуюся в вашей схеме данных таблицу с иерархической структурой (например, pharmacists), написать рекурсивный запрос с помощью WITH RECURSIVE для вывода всей иерархии с указанием уровня вложенности.

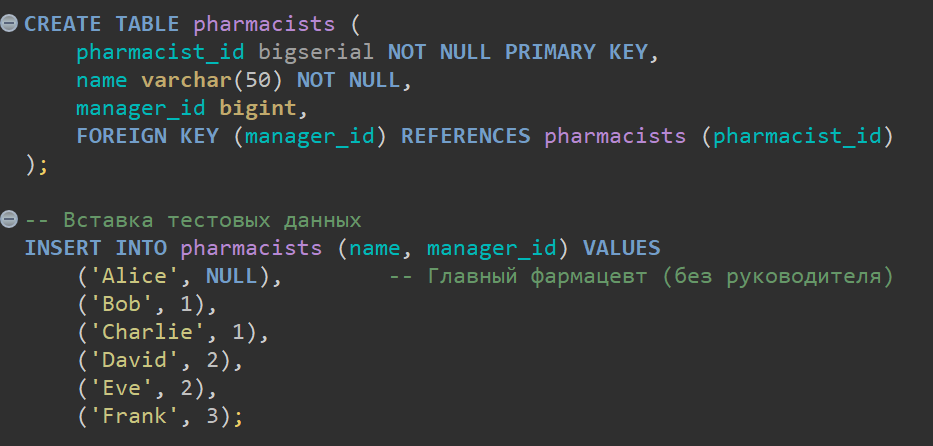


Рисунок 17 – создание таблицы для запроса 3.2

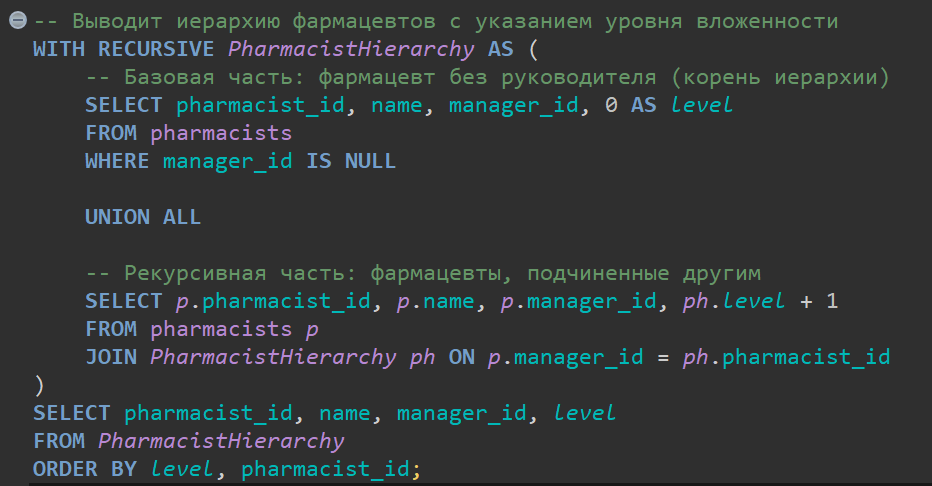


Рисунок 18 – запрос 3.2

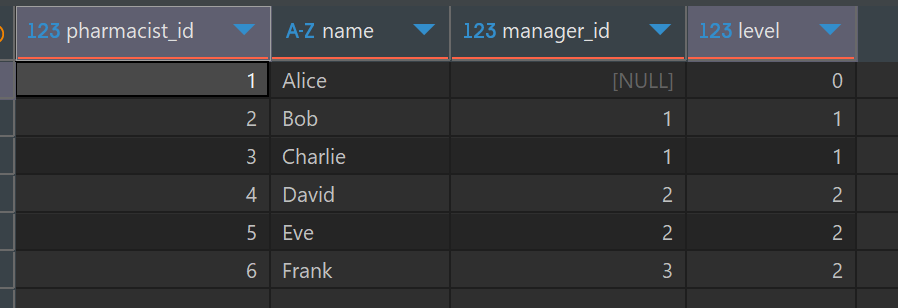


Рисунок 19 – результат запроса 3.2

**ВЫВОД**

Сформированы и закреплены фундаментальные навыки работы с реляционными базами данных на примере СУБД Postgres Pro.