Министерство науки и высшего образования Российской Федерации

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №8

по дисциплине «Системы управления базами данных»

Выполнил:

студент группы 932323

Пронин Л.С.

Проверил:

Преподаватель

Мокина Е. Е.

Томск – 2025

**Задание 1. Вывести все сотрудников (Last\_name F. - departmetn\_name), у которых есть 2 вышестоящих руководителя. Добавить столбец, в котором перечислены эти руководители через запятую.**

WITH RECURSIVE r as (SELECT employee\_id, manager\_id, first\_name, last\_name, 0 as level,

'' as managers, department\_id

FROM employees

WHERE manager\_id is NULL

UNION

SELECT e.employee\_id, e.manager\_id, e.first\_name, e.last\_name , r.level + 1 as level, concat(managers, r.first\_name, ' ', r.last\_name, ', '), e.department\_id

FROM employees e

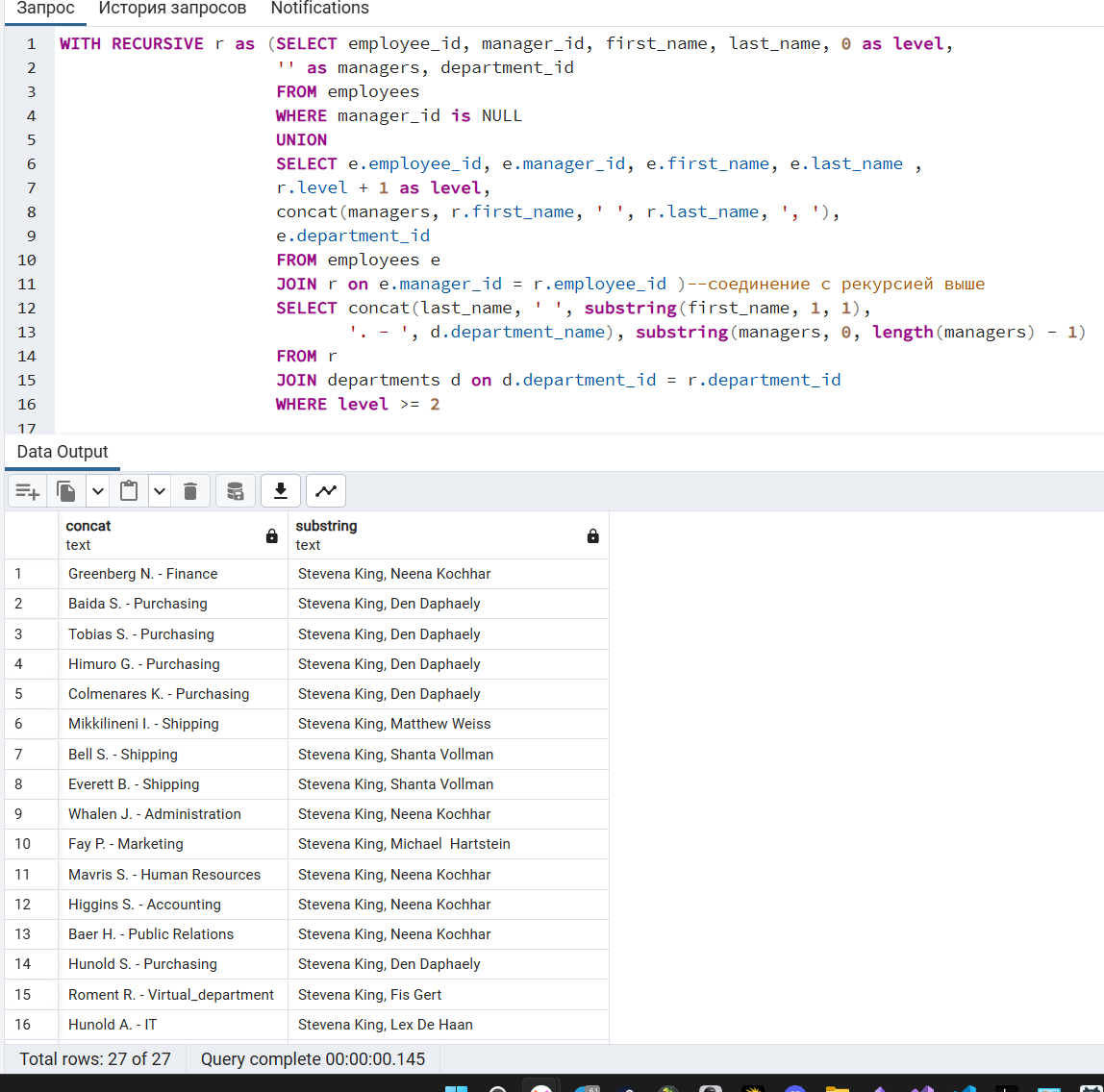
JOIN r on e.manager\_id = r.employee\_id )--соединение с рекурсией выше

SELECT concat(last\_name, ' ', substring(first\_name, 1, 1), '. - ', d.department\_name), substring(managers, 0, length(managers) - 1)

FROM r

JOIN departments d on d.department\_id = r.department\_id

WHERE level >= 2



**Задание 2. Подсчитайте количество подчиненных (всех уровней) для каждого сотрудника, начиная с главного менеджера. При этом отсортировать по кол-ву подчиненных от большего к меньшему и вывести только первых 7 сотрудников, у кого кол-во подчинённых больше 1.**

WITH RECURSIVE r as (SELECT employee\_id, manager\_id, employee\_id as head

FROM employees

UNION

SELECT e.employee\_id, e.manager\_id, r.head as head

FROM employees e

JOIN r on r.employee\_id = e.manager\_id)

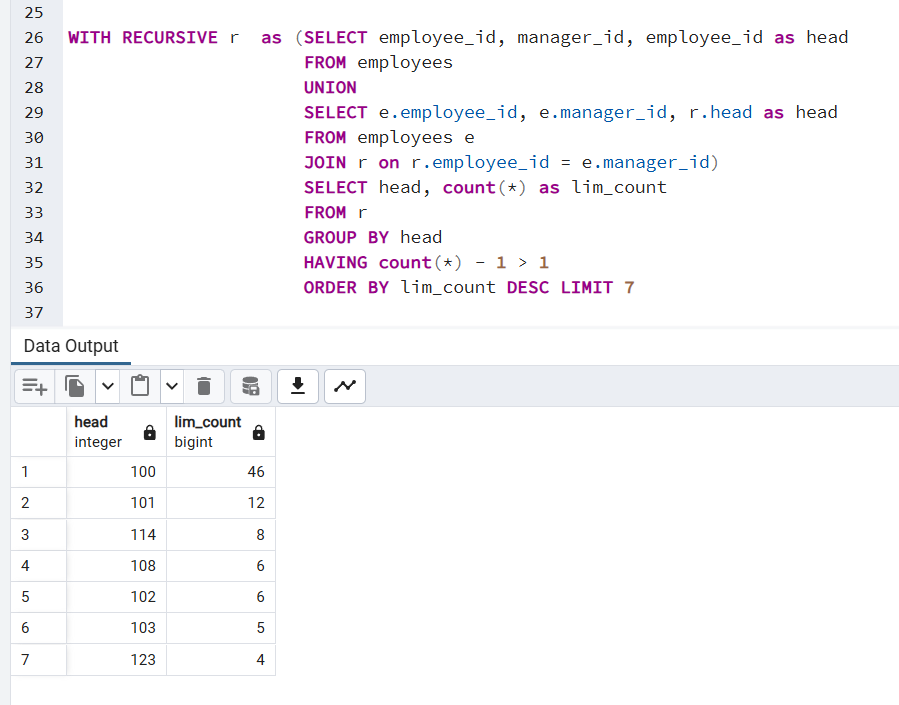
SELECT head, count(\*) as lim\_count

FROM r

GROUP BY head

HAVING count(\*) - 1 > 1

ORDER BY lim\_count DESC LIMIT 7



**Задание 3. Подсчет подчиненных для каждого менеджера**

WITH RECURSIVE r AS (SELECT employee\_id, first\_name, last\_name, manager\_id, 1 as count\_slave

FROM employees

UNION

SELECT e.employee\_id, e.first\_name, e.last\_name, e.manager\_id, r.count\_slave

FROM employees e

JOIN r ON e.manager\_id = r.employee\_id)

SELECT em.first\_name || ' ' || em.last\_name as manager\_name, COUNT(r.employee\_id) as total

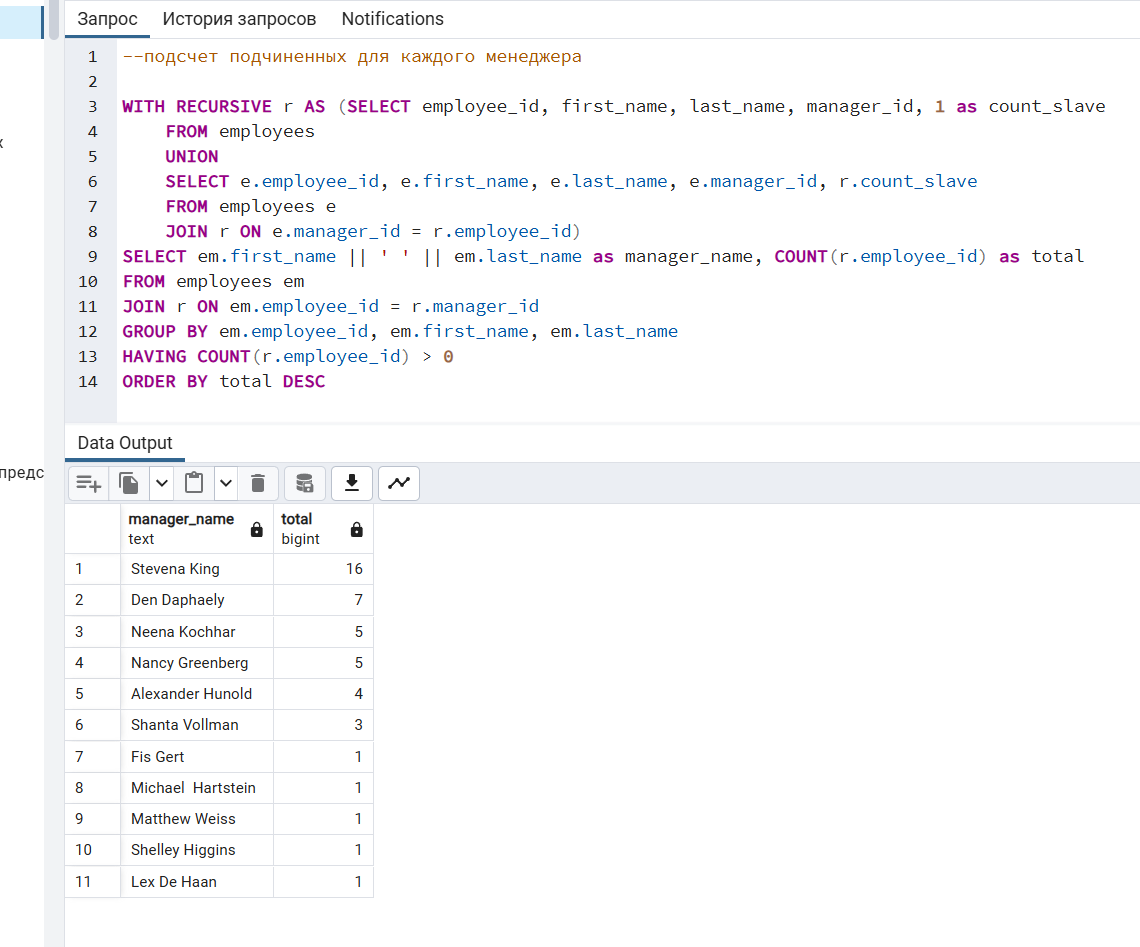
FROM employees em

JOIN r ON em.employee\_id = r.manager\_id

GROUP BY em.employee\_id, em.first\_name, em.last\_name

HAVING COUNT(r.employee\_id) > 0

ORDER BY total DESC



**Защита: вывести всех сотрудников, у которых есть подчиненные только последнего уровня (у которых подчиненные только одного уровня вниз) вывести список подчиненных через запятую и сумму их зарплат**

WITH recursive r as (SELECT employee\_id, first\_name, last\_name, manager\_id, salary, 0 as level

FROM employees

UNION

SELECT e.employee\_id, e.first\_name, e.last\_name, e.manager\_id, e.salary, r.level + 1 as level

FROM employees e

JOIN r on e.manager\_id = r.employee\_id)

SELECT concat(emp.last\_name, ' ', substring(emp.first\_name, 1, 1), '. - ', d.department\_name),

STRING\_AGG(concat(e.first\_name, ' ', e.last\_name), ', '), ROUND(SUM(e.salary)) as sum\_salary

FROM employees emp

JOIN departments d on emp.department\_id = d.department\_id

JOIN employees e ON emp.employee\_id = e.manager\_id

WHERE e.employee\_id NOT IN (SELECT manager\_id

FROM employees

WHERE manager\_id IS NOT NULL)--тут мы исключаем NULL

GROUP BY emp.employee\_id, emp.first\_name, emp.last\_name, d.department\_name

ORDER BY sum\_salary DESC;