Министерство науки и высшего образования Российской Федерации

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

по дисциплине «Системы управления базами данных»

«Соединение таблиц»

Выполнил: Левковский А. Д.

студент группы 932323

Проверил:

Преподаватель

Мокина Е. Е.

Томск – 2024

-- 1 –

Напишите запрос для отображения названия отдела и количества сотрудников в каждом из отделов и средней заработной платой в отделе, где средняя ЗП больше 10000 (можете подобрать другое число)

--SELECT d.department\_name,

--COUNT(e.employee\_id) AS employee\_cnt,

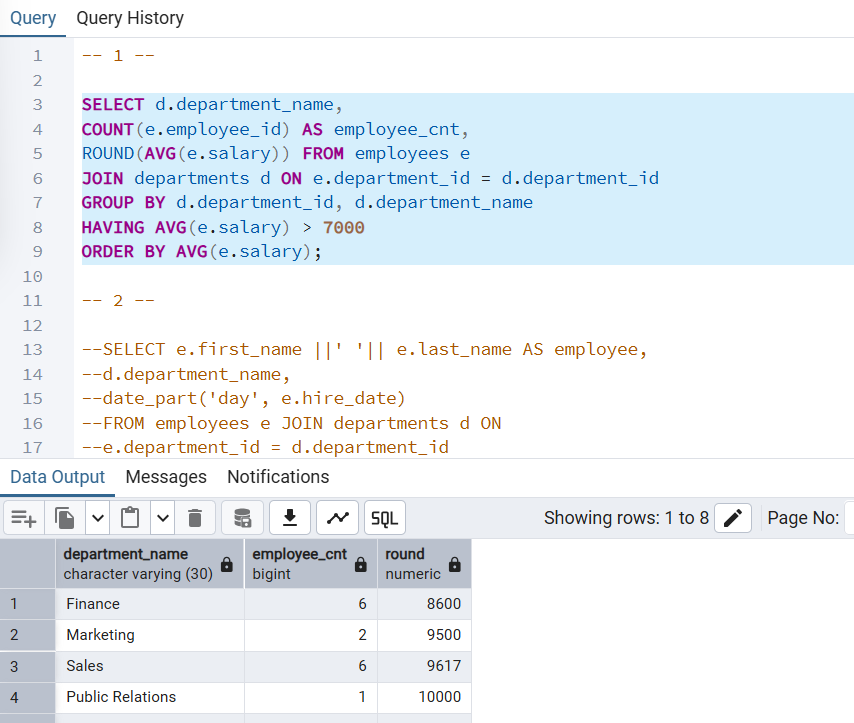
--ROUND(AVG(e.salary)) FROM employees e

--JOIN departments d ON e.department\_id = d.department\_id

--GROUP BY d.department\_id, d.department\_name

--HAVING AVG(e.salary) > 7000

--ORDER BY AVG(e.salary);



-- 2 --

Напишите запрос для отображения полного имени (имени и фамилии), названия должности, даты начала для тех сотрудников, у которых не указан номер телефона.

--SELECT e.first\_name ||' '|| e.last\_name AS employee,

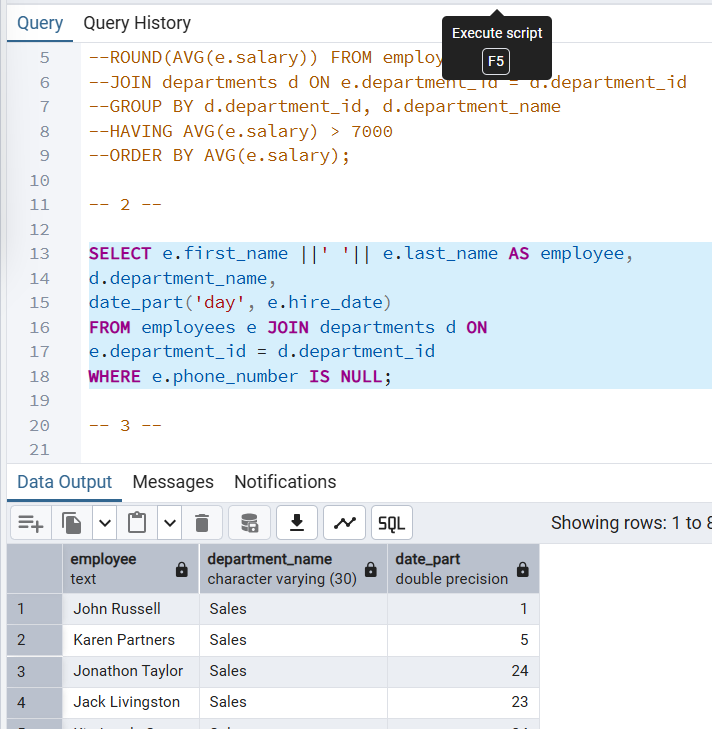
--d.department\_name,

--date\_part('day', e.hire\_date)

--FROM employees e JOIN departments d ON

--e.department\_id = d.department\_id

--WHERE e.phone\_number IS NULL;



-- 3 --

Напишите запрос для отображения идентификатора сотрудника, названия должности, в отделе с названием IT, у которых заработная плата более 5000.

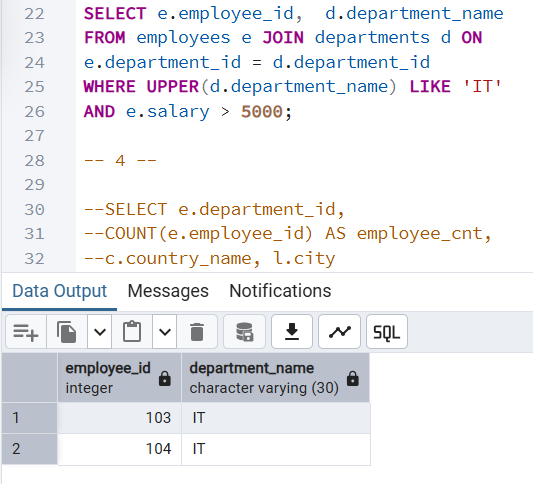
--SELECT e.employee\_id, d.department\_name

--FROM employees e JOIN departments d ON

--e.department\_id = d.department\_id

--WHERE UPPER(d.department\_name) LIKE 'IT'

--AND e.salary > 5000;



-- 4 --

Напишите запрос для вывода всех отделов, в которых работают минимум 2 сотрудника, страну и город этих отделов, даже если они пустые.

-- SELECT e.department\_id,

-- COUNT(e.employee\_id) AS employee\_cnt,

-- c.country\_name, l.city

-- FROM employees e Right JOIN departments d ON

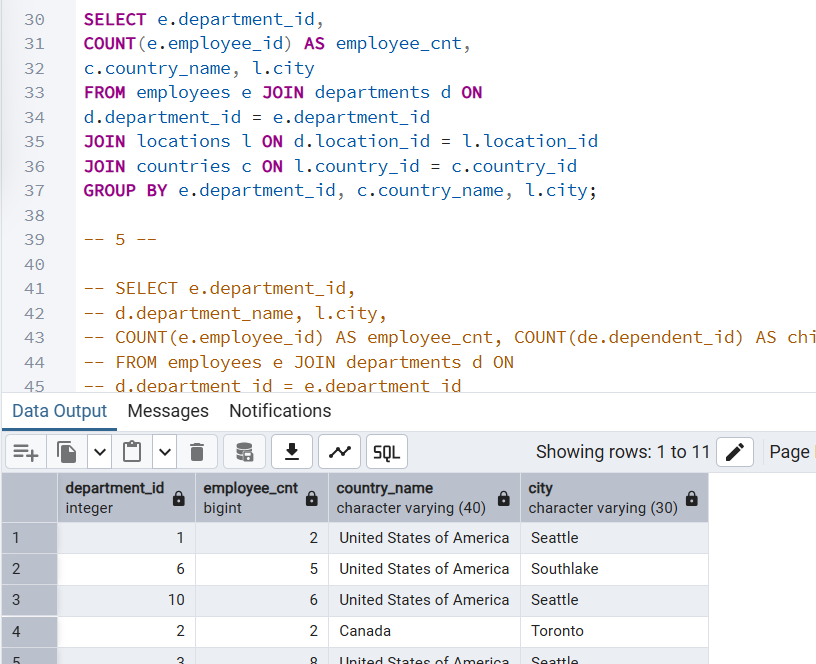
-- d.department\_id = e.department\_id

-- left JOIN locations l ON d.location\_id = l.location\_id

-- left JOIN countries c ON l.country\_id = c.country\_id

-- GROUP BY e.department\_id, c.country\_name, l.city

-- HAVING COUNT(e.employee\_id) > 1;



-- 5 --

Напишите запрос, чтобы отобразить название отдела и город, кол-во сотрудников этого отдела и кол-во детей всех сотрудников этого отдела. Внимательнее: у некоторых сотрудников нет детей.

-- SELECT e.department\_id,

-- d.department\_name, l.city,

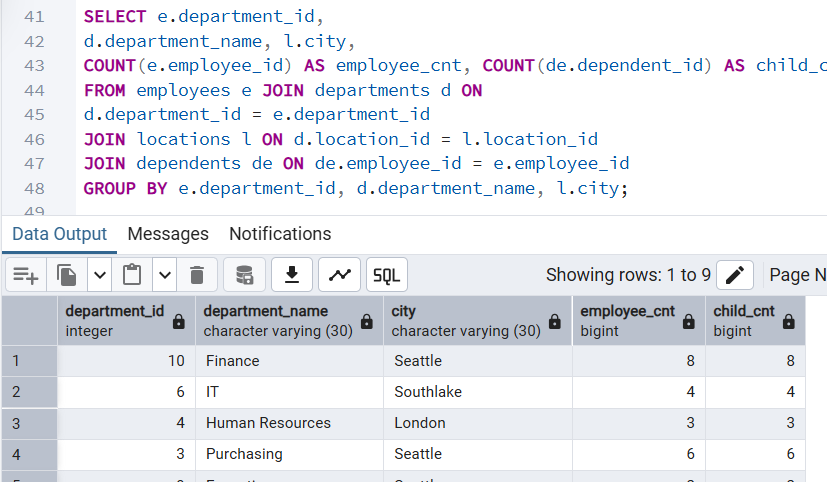
-- COUNT(e.employee\_id) AS employee\_cnt, COUNT(de.dependent\_id) AS child\_cnt

-- FROM employees e right JOIN departments d ON

-- d.department\_id = e.department\_id

-- left JOIN locations l ON d.location\_id = l.location\_id

-- left JOIN dependents de ON de.employee\_id = e.employee\_id

-- GROUP BY e.department\_id, d.department\_name, l.city;

-- 6 --

Для каждого сотрудника укажите полное имя, название отдела, город и регион. Отсортируйте по Фамилии сотрудника в обратном порядке.

-- SELECT e.last\_name ||' '|| e.first\_name AS employee, d.department\_name,

-- l.city, r.region\_name

-- FROM employees e left JOIN departments d ON

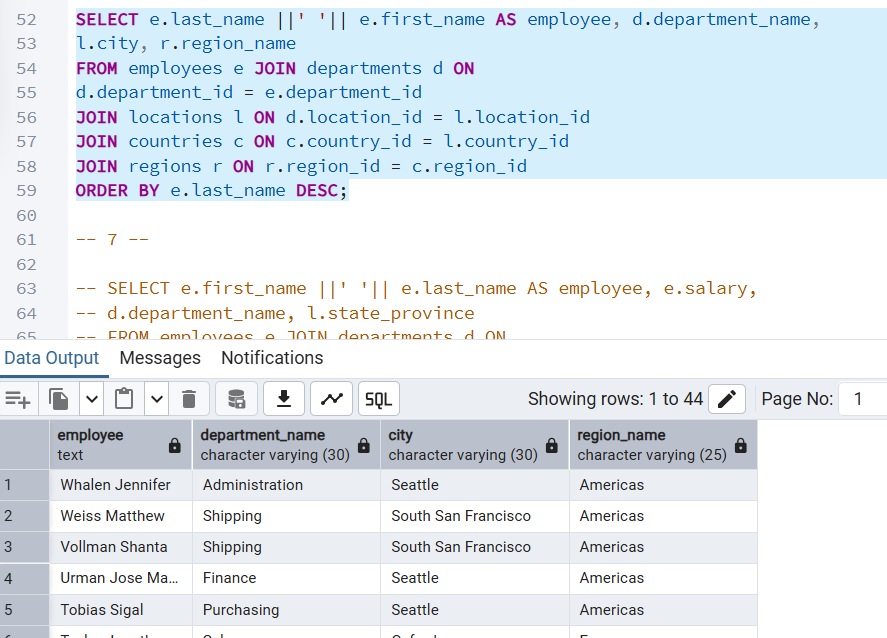
-- d.department\_id = e.department\_id

-- left JOIN locations l ON d.location\_id = l.location\_id

-- left JOIN countries c ON c.country\_id = l.country\_id

-- left JOIN regions r ON r.region\_id = c.region\_id

-- ORDER BY e.last\_name DESC;



-- 7 --

Написать запрос для вывода полного имени (имени и фамилии) и зарплаты сотрудников и названия всех отделов, расположенных в штате Вашингтон.

-- SELECT e.first\_name ||' '|| e.last\_name AS employee, e.salary,

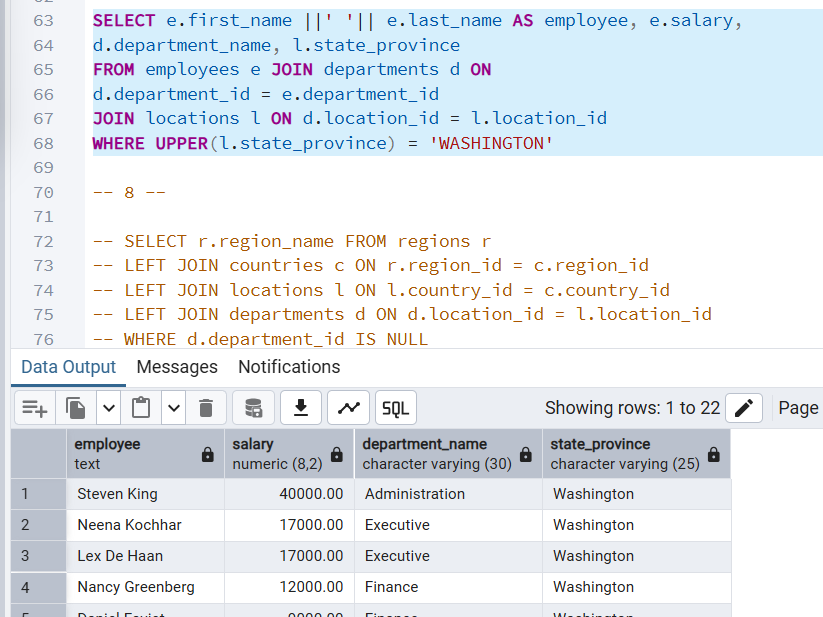
-- d.department\_name, l.state\_province

-- FROM employees e JOIN departments d ON

-- d.department\_id = e.department\_id

-- JOIN locations l ON d.location\_id = l.location\_id

-- WHERE UPPER(l.state\_province) = 'WASHINGTON'



-- 8 --

Выведите регионы, в которых нет ни одного отдела.

SELECT r.region\_name, COUNT(d.department\_id) AS cnt FROM regions r

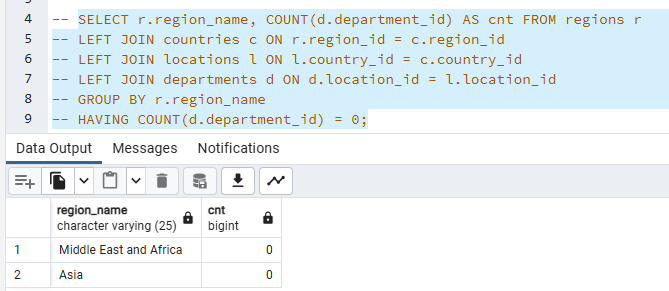
-- LEFT JOIN countries c ON r.region\_id = c.region\_id

-- LEFT JOIN locations l ON l.country\_id = c.country\_id

-- LEFT JOIN departments d ON d.location\_id = l.location\_id

-- GROUP BY r.region\_name

-- HAVING COUNT(d.department\_id) = 0;



-- 9 --

Отобразите полное имя, название отдела, город и регион для всех сотрудников, фамилия которых содержит букву «o».

-- SELECT e.last\_name ||' '|| e.first\_name AS employee, d.department\_name,

-- l.city, r.region\_name

-- FROM employees e LEFT JOIN departments d ON

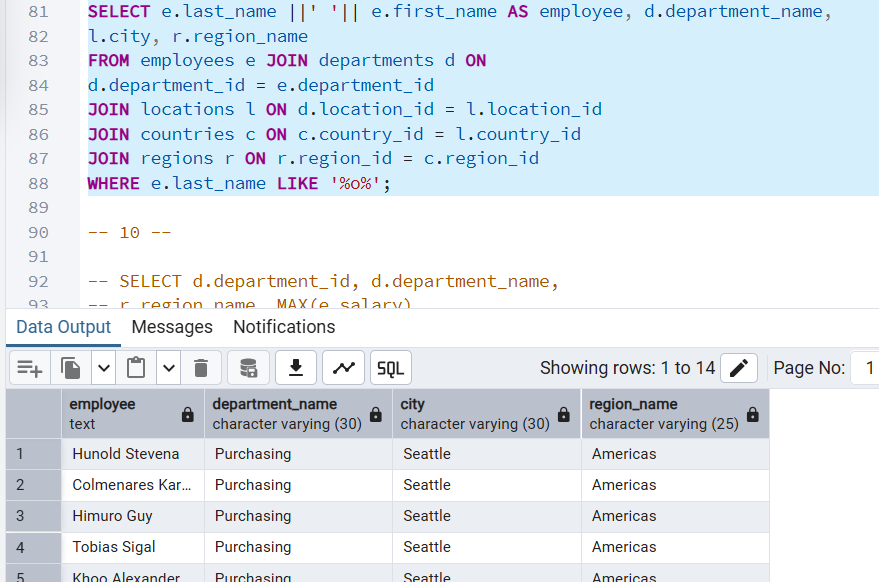
-- d.department\_id = e.department\_id

-- LEFT JOIN locations l ON d.location\_id = l.location\_id

-- LEFT JOIN countries c ON c.country\_id = l.country\_id

-- LEFT JOIN regions r ON r.region\_id = c.region\_id

-- WHERE e.last\_name LIKE '%o%';



-- 10 --

Для каждого отдела отобразите номер отдела, название отдела, регион и максимальную ЗП в отделе и список сотрудников в виде "Фамилия И." разделенных через запятую.

-- SELECT d.department\_id, d.department\_name,

-- r.region\_name, MAX(e.salary),

-- STRING\_AGG(e.last\_name ||' '|| UPPER(SUBSTR(e.first\_name, 1, 1)), ', ') AS employee

-- FROM employees e right JOIN departments d ON

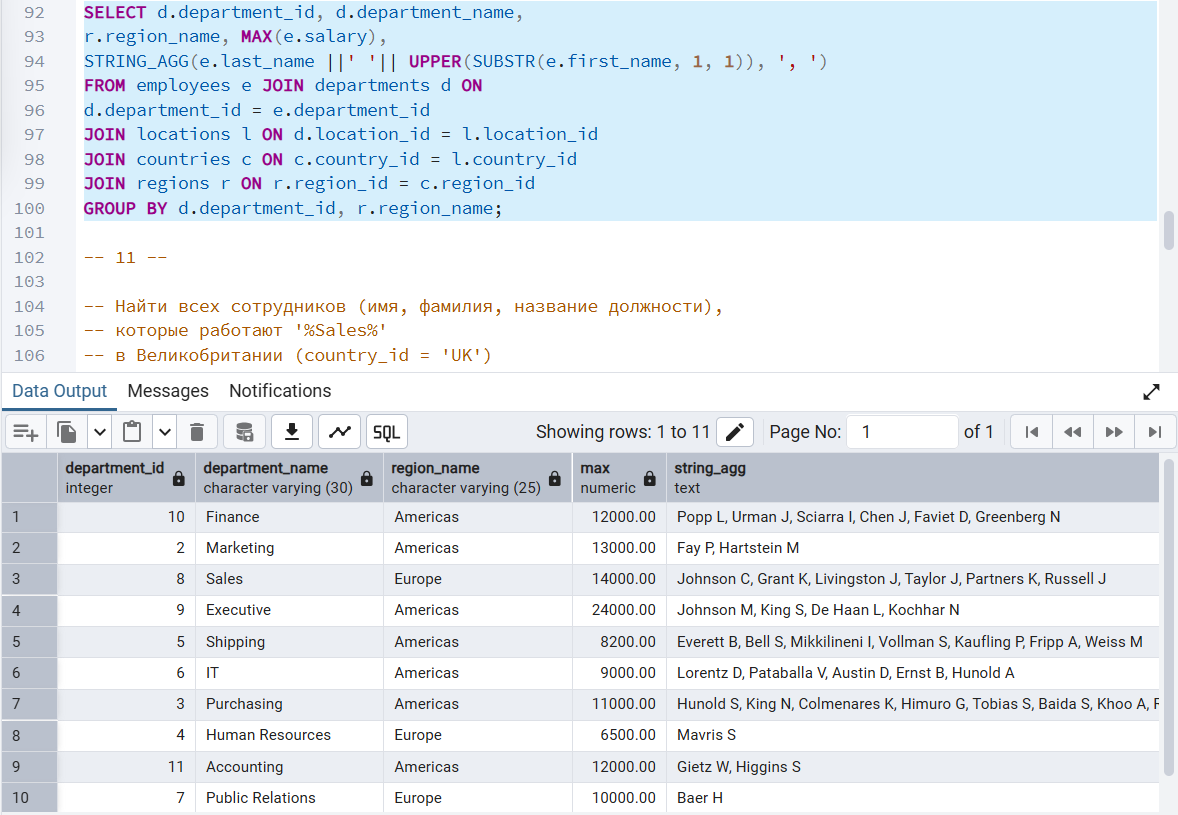
-- d.department\_id = e.department\_id

-- left JOIN locations l ON d.location\_id = l.location\_id

-- left JOIN countries c ON c.country\_id = l.country\_id

-- left JOIN regions r ON r.region\_id = c.region\_id

-- GROUP BY d.department\_id, r.region\_name;



-- 11 --

-- Найти всех сотрудников (имя, фамилия, название должности),

-- которые работают '%Sales%'

-- в Великобритании (country\_id = 'UK')

-- и у которых есть дети

-- SELECT e.first\_name ||' '|| e.last\_name AS employee,

-- d.department\_name FROM employees e

-- JOIN departments d ON d.department\_id = e.department\_id

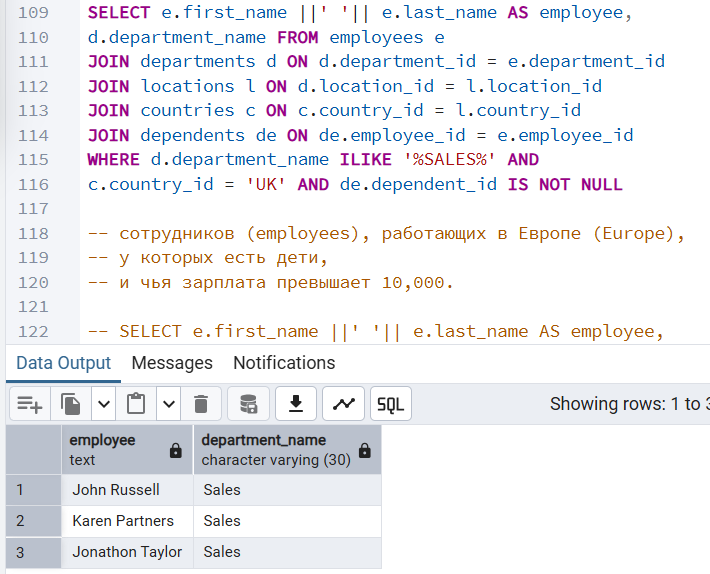
-- JOIN locations l ON d.location\_id = l.location\_id

-- JOIN countries c ON c.country\_id = l.country\_id

-- JOIN dependents de ON de.employee\_id = e.employee\_id

-- WHERE d.department\_name ILIKE '%SALES%' AND

-- c.country\_id = 'UK' AND de.dependent\_id IS NOT NULL



-- 12 --

-- Проанализировать эффективность локаций для отделов. Мы хотим увидеть

-- ВСЕ локации (даже те, где нет отделов), а также ВСЕ отделы

-- (даже те, которые не привязаны к локациям), и для тех отделов,

-- где есть сотрудники, вывести информацию о них.

-- SELECT l.city, d.department\_name,

-- e.first\_name ||' '|| e.last\_name AS employee, e.salary

-- FROM departments d

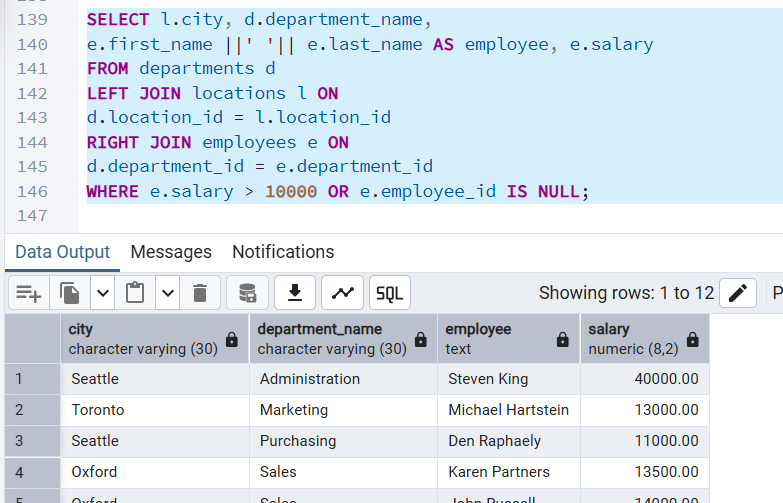
-- LEFT JOIN locations l ON

-- d.location\_id = l.location\_id

-- RIGHT JOIN employees e ON

-- d.department\_id = e.department\_id

-- WHERE e.salary > 10000 OR e.employee\_id IS NULL;



Защита: