Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

Лабораторная работа № 1

“Вариант 3”

Отчет

Выполнил:

студент гр. 221703

Веркович Е.В.

Проверил: Ерофеев И.А.

Минск 2024

**Задание 2. Составить последовательность реляционных операций для задач по варианту.**

- Инициализация таблиц:

CREATE TABLE suppliers (

    id SERIAL PRIMARY KEY NOT NULL,

    name VARCHAR(30) NOT NULL,

    status INTEGER NOT NULL,

    city VARCHAR(30) NOT NULL

);

CREATE TABLE products (

    id SERIAL PRIMARY KEY NOT NULL,

    name VARCHAR(30) NOT NULL,

    color VARCHAR(30) NOT NULL,

    size INTEGER NOT NULL,

    city VARCHAR(30) NOT NULL

);

CREATE TABLE projects (

    id SERIAL PRIMARY KEY NOT NULL,

    name VARCHAR(30) NOT NULL,

    city VARCHAR(30) NOT NULL

);

CREATE TABLE quantities (

    supplier\_id INTEGER NOT NULL,

    product\_id INTEGER NOT NULL,

    project\_id INTEGER NOT NULL,

    quantity INTEGER NOT NULL,

    FOREIGN KEY (supplier\_id) REFERENCES suppliers(id),

    FOREIGN KEY (product\_id) REFERENCES products(id),

    FOREIGN KEY (product\_id) REFERENCES products(id)

);

-Заполнение таблиц данными:

INSERT INTO suppliers(name, status, city)

VALUES

    ('Петров', 20, 'Москва'),

    ('Синицин', 10, 'Таллинн'),

    ('Федоров', 30, 'Таллинн'),

    ('Чаянов', 20, 'Минск'),

    ('Крюков', 30, 'Киев');

INSERT INTO products(name, color, size, city)

VALUES

    ('Болт', 'Красный', 12, 'Москва'),

    ('Гайка', 'Зеленый', 17, 'Минск'),

    ('Диск', 'Черный', 17, 'Вильнюс'),

    ('Диск', 'Черный', 14, 'Москва'),

    ('Корпус', 'Красный', 12, 'Минск'),

    ('Крышки', 'Красный', 19, 'Москва');

INSERT INTO projects(name, city)

VALUES

    ('ИПР1', 'Минск'),

    ('ИПР2', 'Таллинн'),

    ('ИПР3', 'Псков'),

    ('ИПР4', 'Псков'),

    ('ИПР5', 'Москва'),

    ('ИПР6', 'Саратов'),

    ('ИПР7', 'Москва');

INSERT INTO quantities(supplier\_id, product\_id, project\_id, quantity)

VALUES

    (1, 1, 1, 200),

    (1, 1, 2, 700),

    (2, 3, 1, 400),

    (2, 2, 2, 200),

    (2, 3, 3, 200),

    (2, 3, 4, 500),

    (2, 3, 5, 600),

    (2, 3, 6, 400),

    (2, 3, 7, 800),

    (2, 5, 2, 100),

    (3, 3, 1, 200),

    (3, 4, 2, 500),

    (4, 6, 3, 300),

    (4, 6, 7, 300),

    (5, 2, 2, 200),

    (5, 2, 4, 100),

    (5, 5, 5, 500),

    (5, 5, 7, 100),

    (5, 6, 2, 200),

    (5, 1, 2, 100),

    (5, 3, 4, 200),

    (5, 4, 4, 800),

    (5, 5, 4, 400),

    (5, 6, 4, 500);

Таблица поставщиков:

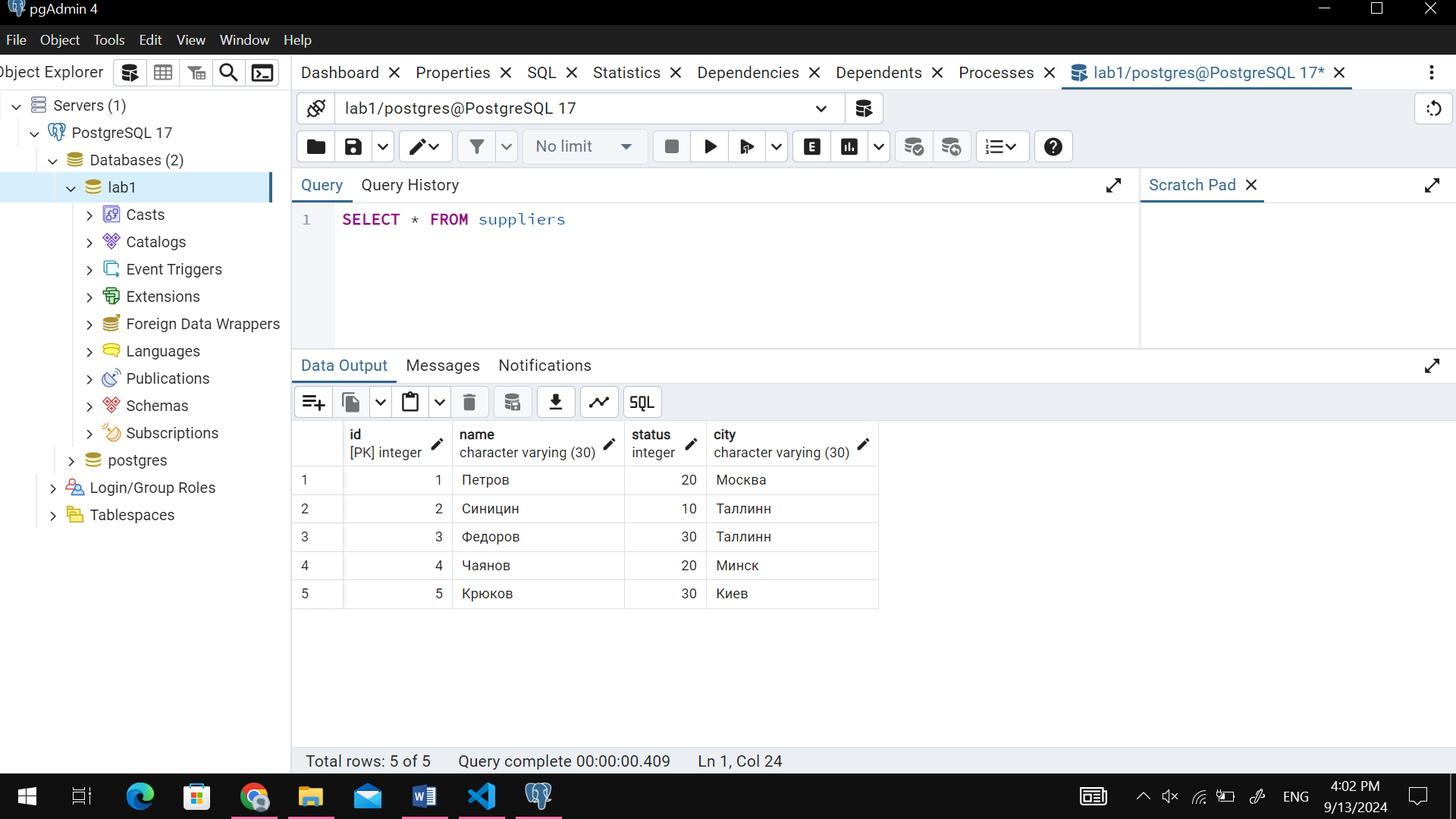


Таблица деталей:

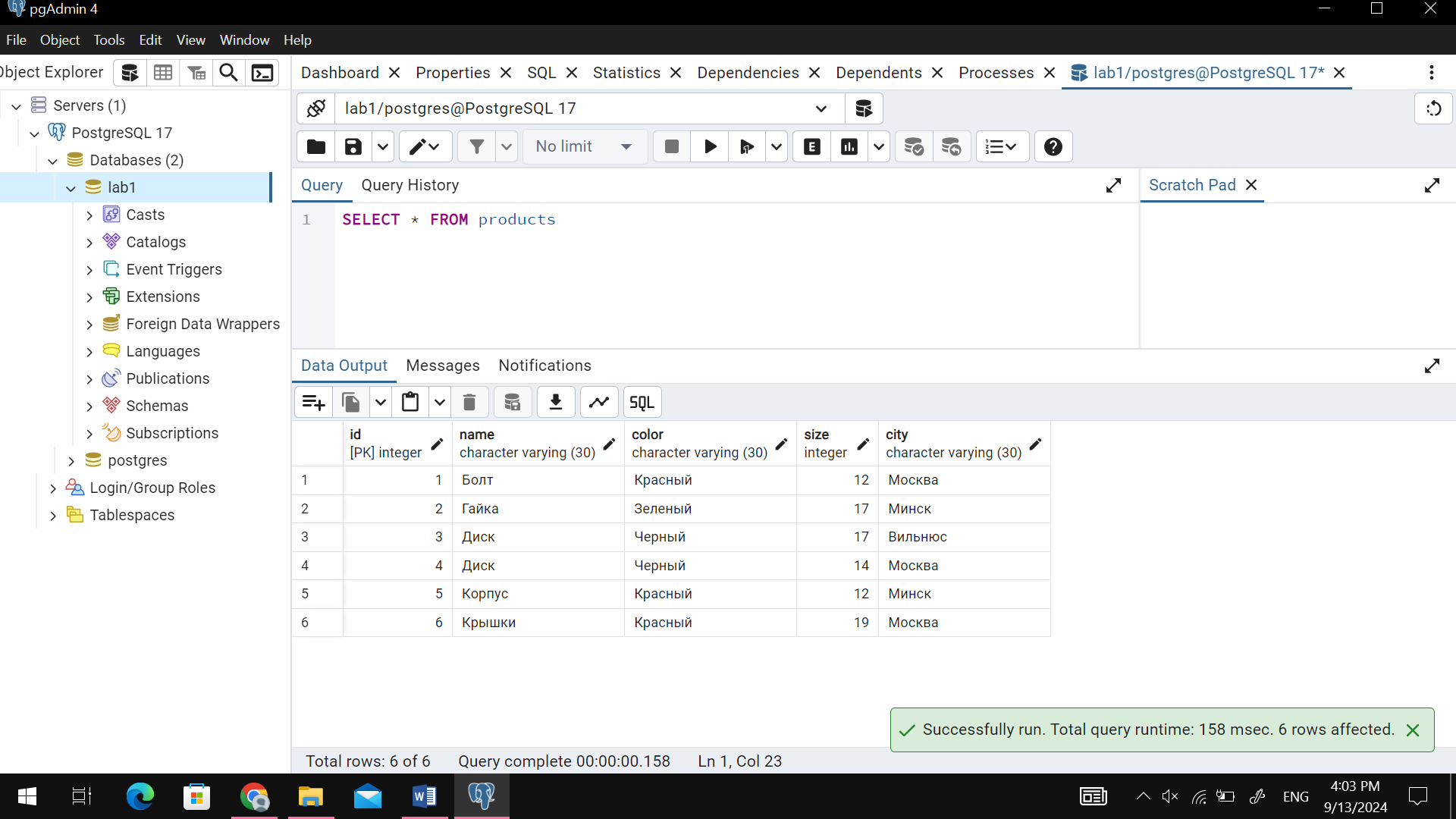


Таблица проектов:

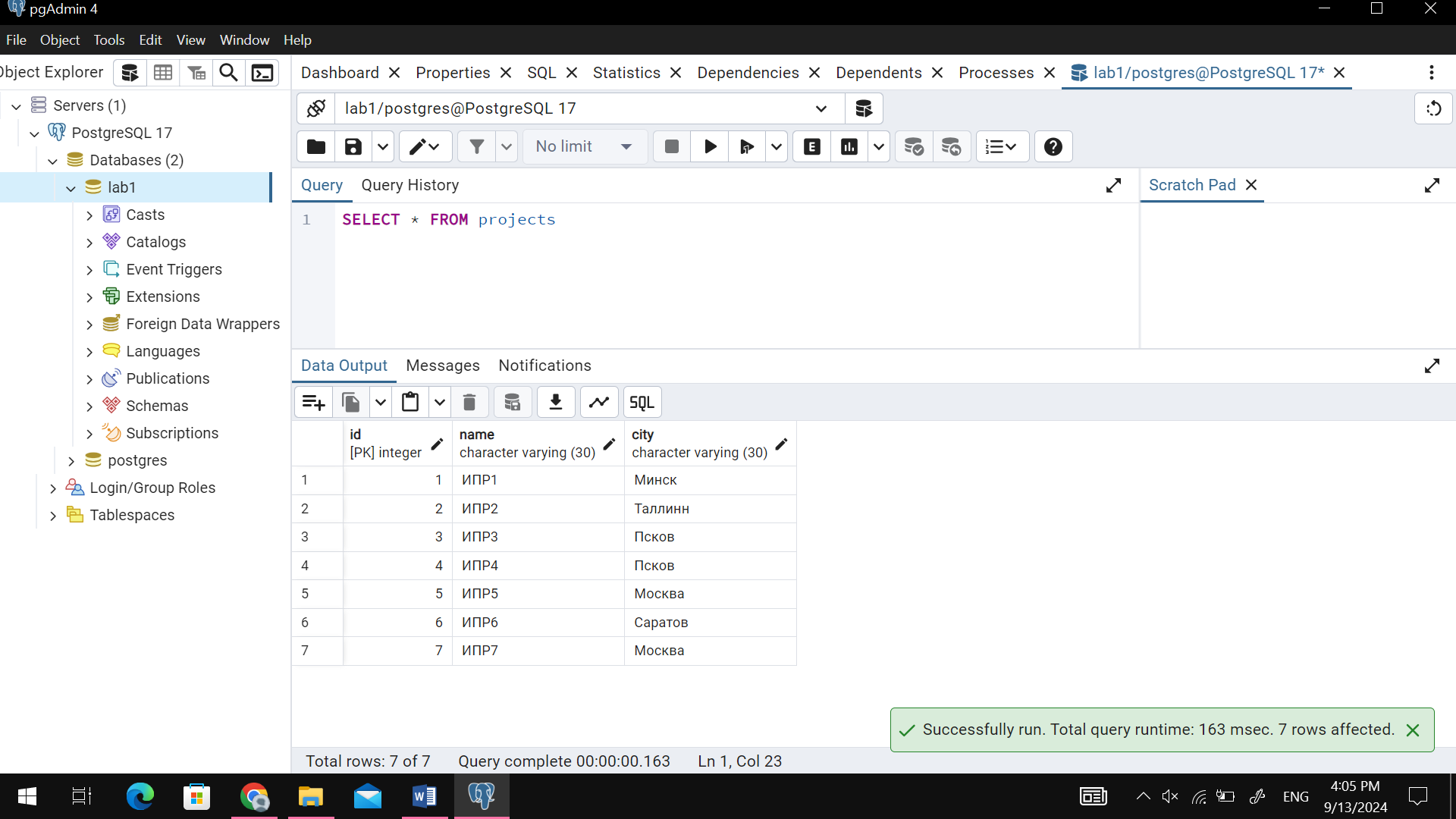
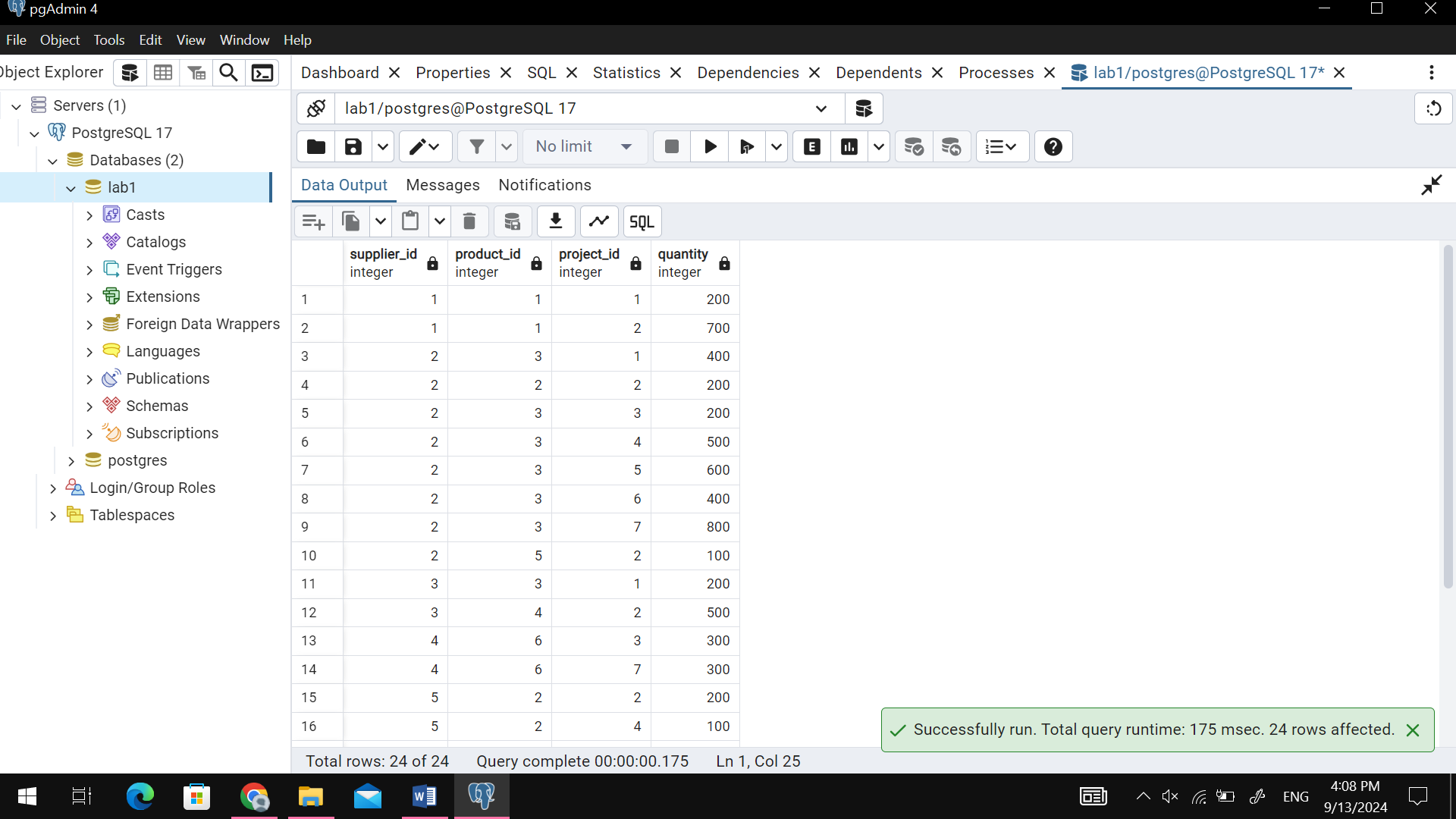
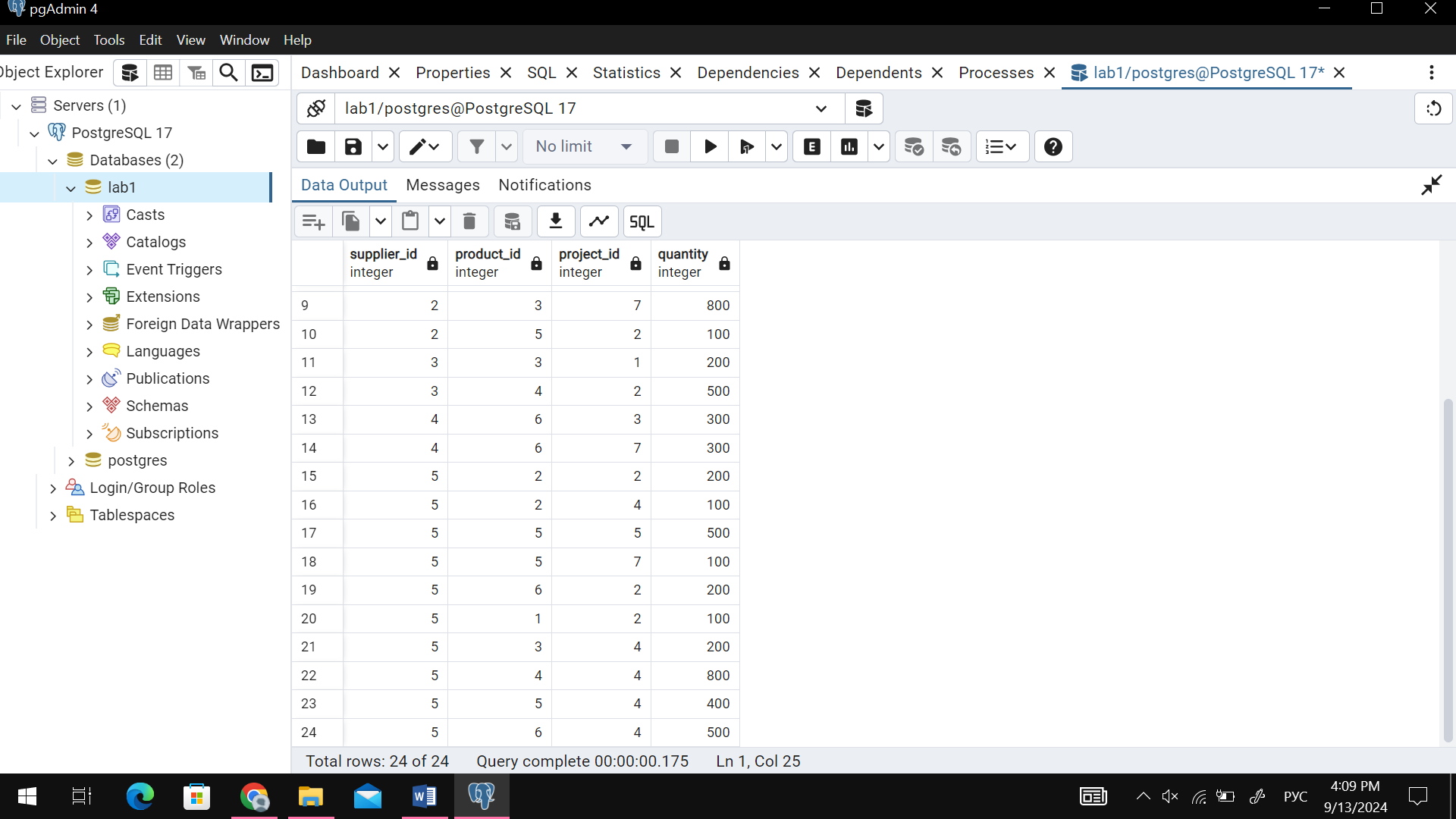


Таблица количества деталей поставляемых проектам от каждого поставщика:





**Запросы**

**Задача №30: Получить номера деталей, поставляемых для лондонских проектов.**

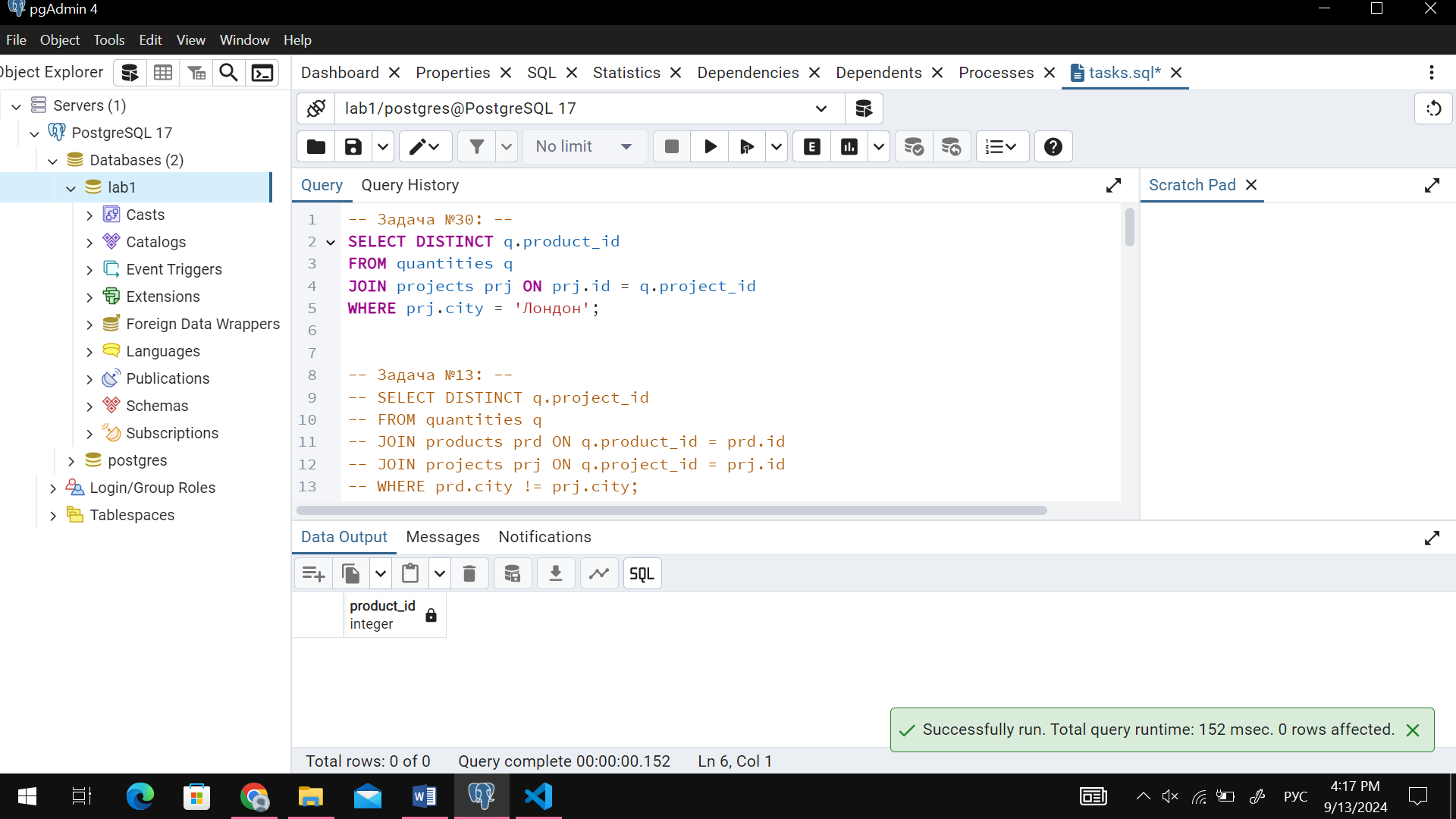
SELECT DISTINCT q.product\_id

FROM quantities q

JOIN projects prj ON prj.id = q.project\_id

WHERE prj.city = 'Лондон';

Результат:



**Задача №13: Получить номера проектов, обеспечиваемых по крайней мере одним поставщиком не из того же города.**

SELECT DISTINCT q.project\_id

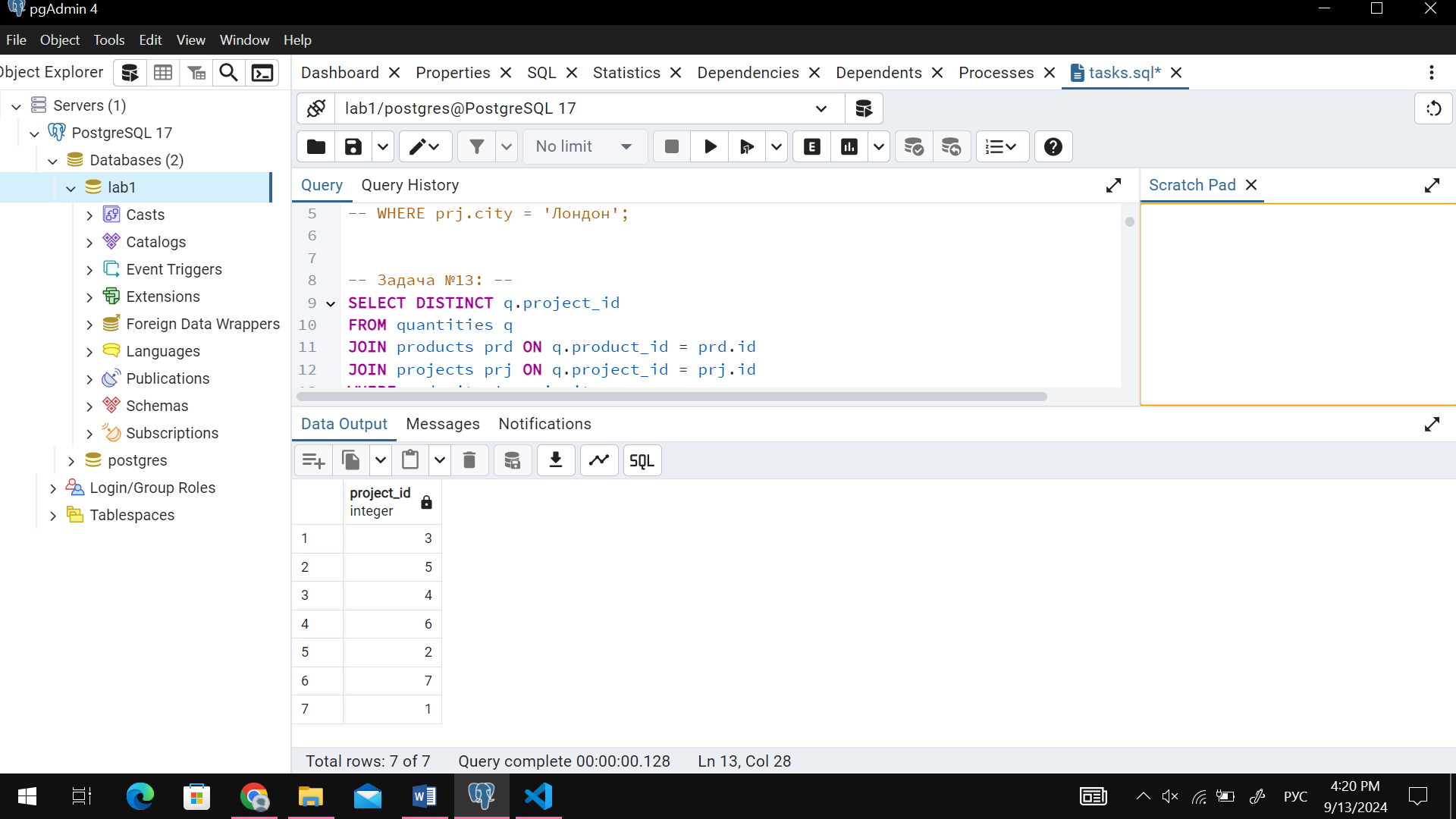
FROM quantities q

JOIN products prd ON q.product\_id = prd.id

JOIN projects prj ON q.project\_id = prj.id

WHERE prd.city != prj.city;

**Результат:**



**Задача №21: Получить номера деталей, поставляемых для какого-либо проекта в Лондоне.**

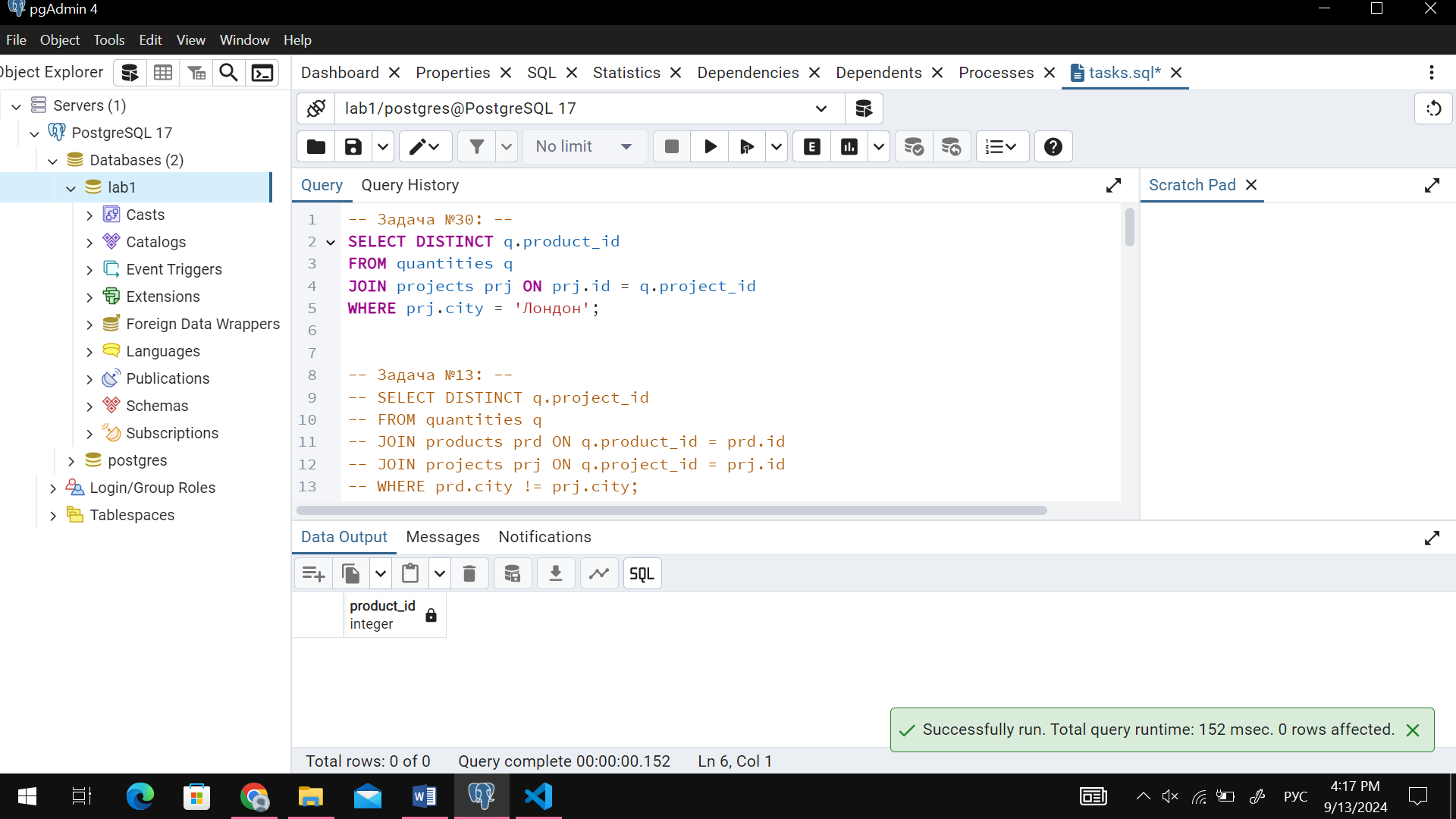
SELECT DISTINCT q.product\_id

FROM quantities q

JOIN projects prj ON prj.id = q.project\_id

WHERE prj.city = 'Лондон';

**Результат:**



**Задача №12: Получить номера деталей, поставляемых для всех проектов, обеспечиваемых поставщиком из того же города, где размещен проект.**

SELECT q.product\_id, s.city, p.city

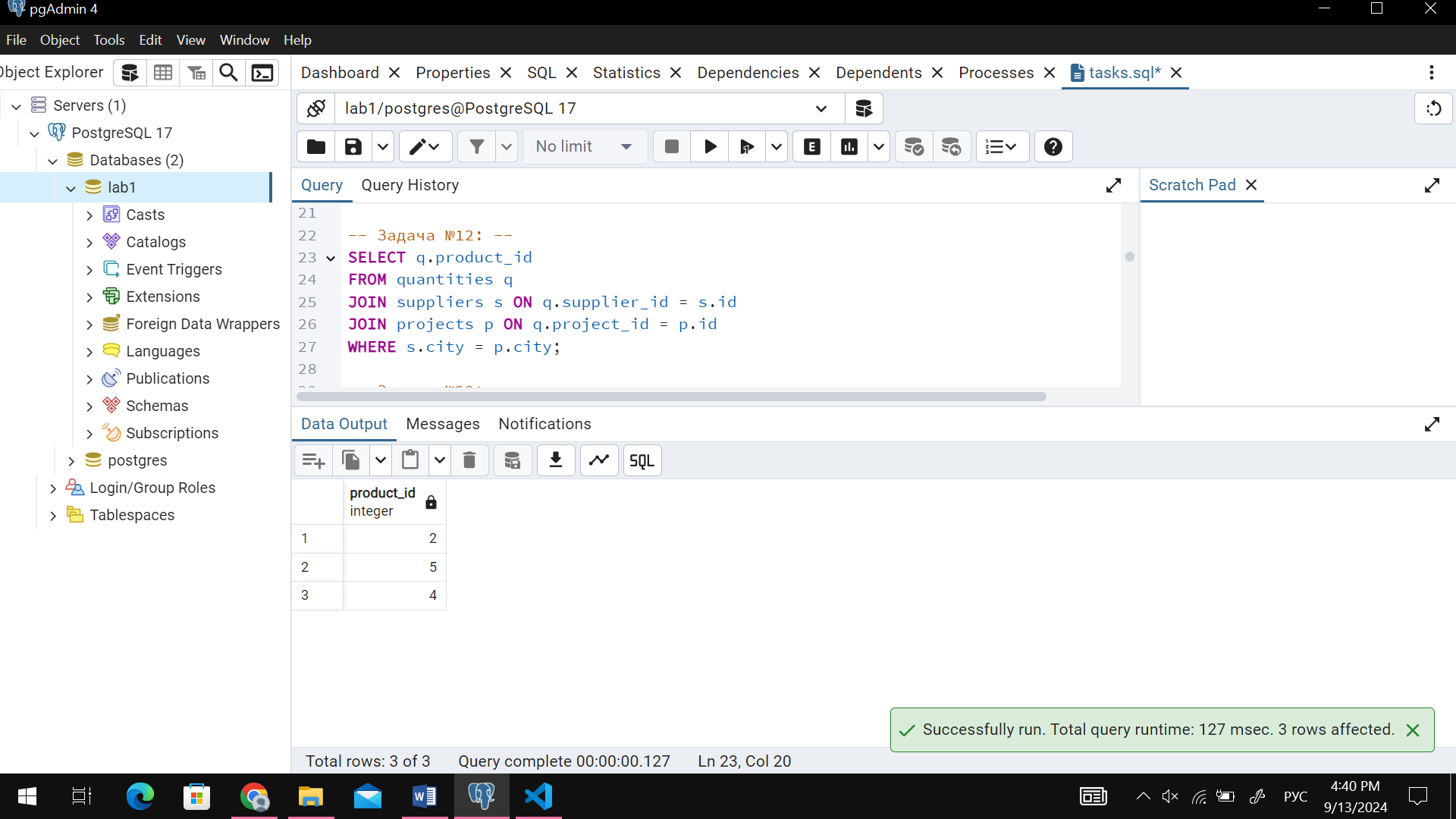
FROM quantities q

JOIN suppliers s ON q.supplier\_id = s.id

JOIN projects p ON q.project\_id = p.id

WHERE s.city = p.city;

**Результат:**



**Задача №28: Получить номера проектов, для которых не поставляются красные детали поставщиками из Лондона.**

SELECT prj.id

FROM projects prj

WHERE prj.id NOT IN

    (SELECT q.project\_id

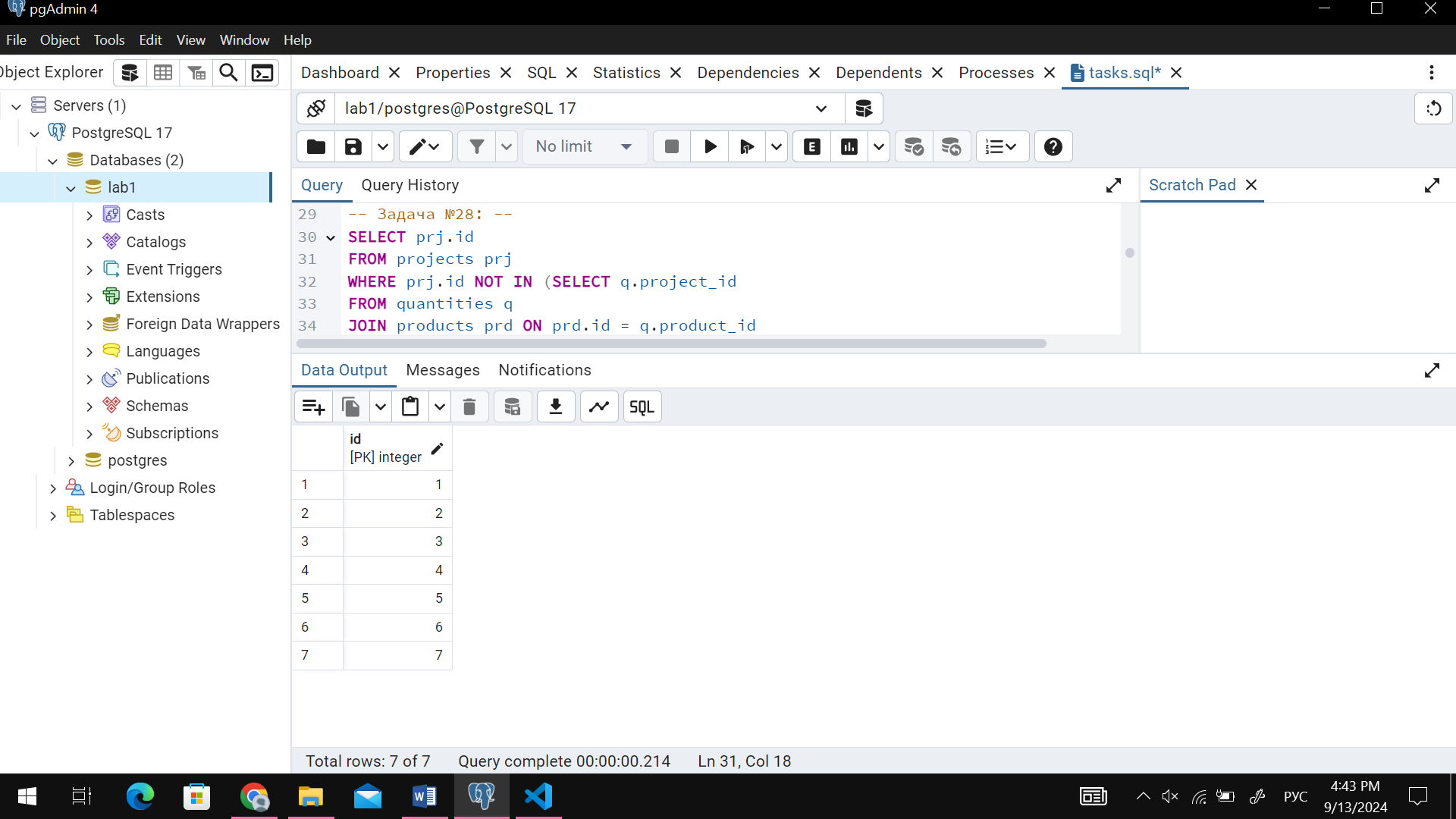
    FROM quantities q

    JOIN products prd ON prd.id = q.product\_id

    JOIN suppliers s ON s.id = q.supplier\_id

    WHERE s.city = 'Лондон' AND prd.color = 'Красный');

**Результат:**



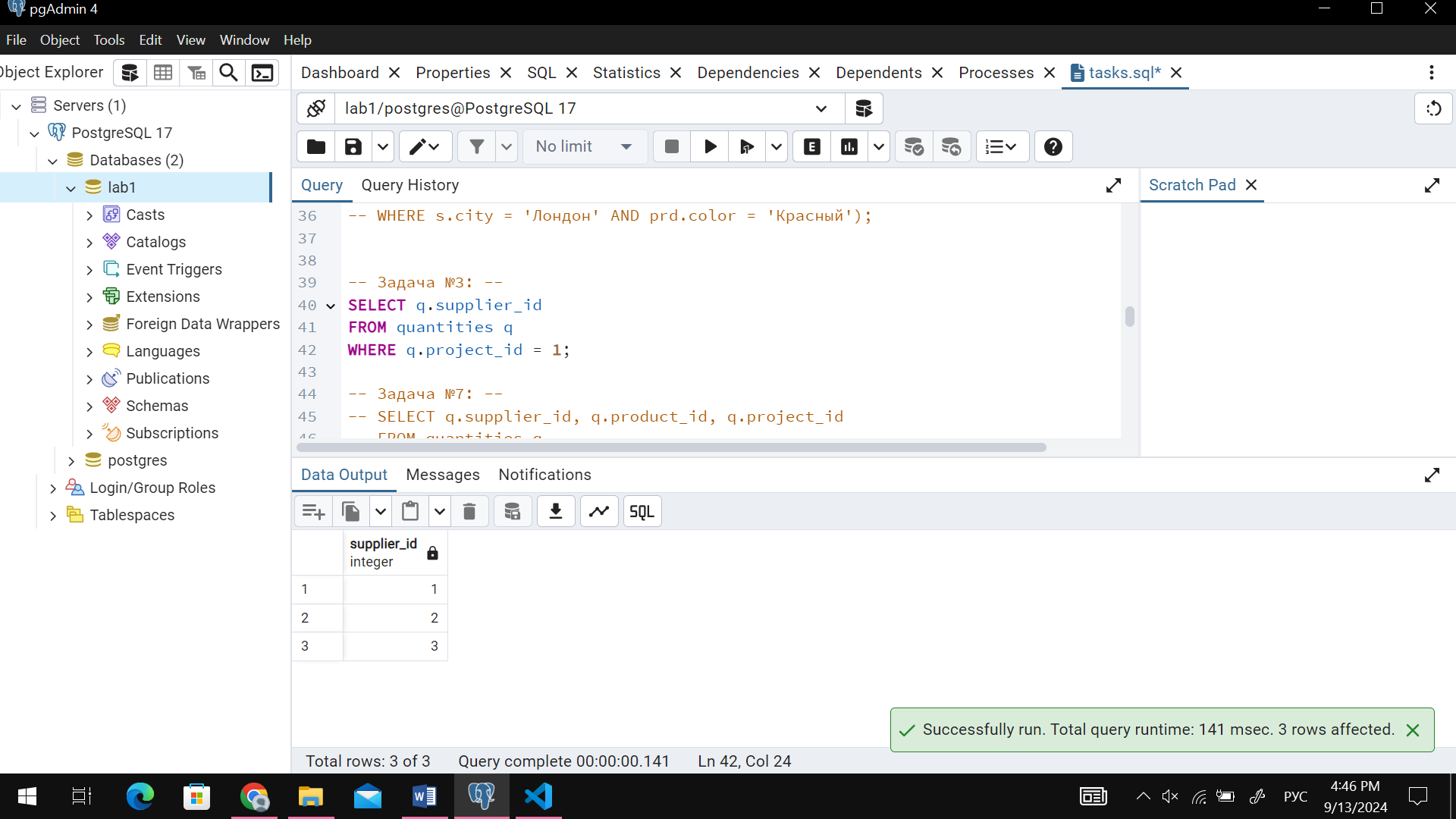
**Задача №3: Получить номера поставщиков, которые обеспечивают проект ПР1.**

SELECT q.supplier\_id

FROM quantities q

WHERE q.project\_id = 1;

**Результат:**



**Задача №7: Получить все такие тройки "номера поставщиков-номера деталей-номера проектов", для которых выводимые поставщик, деталь и проект не размещены в одном городе.**

SELECT q.supplier\_id, q.product\_id, q.project\_id

FROM quantities q

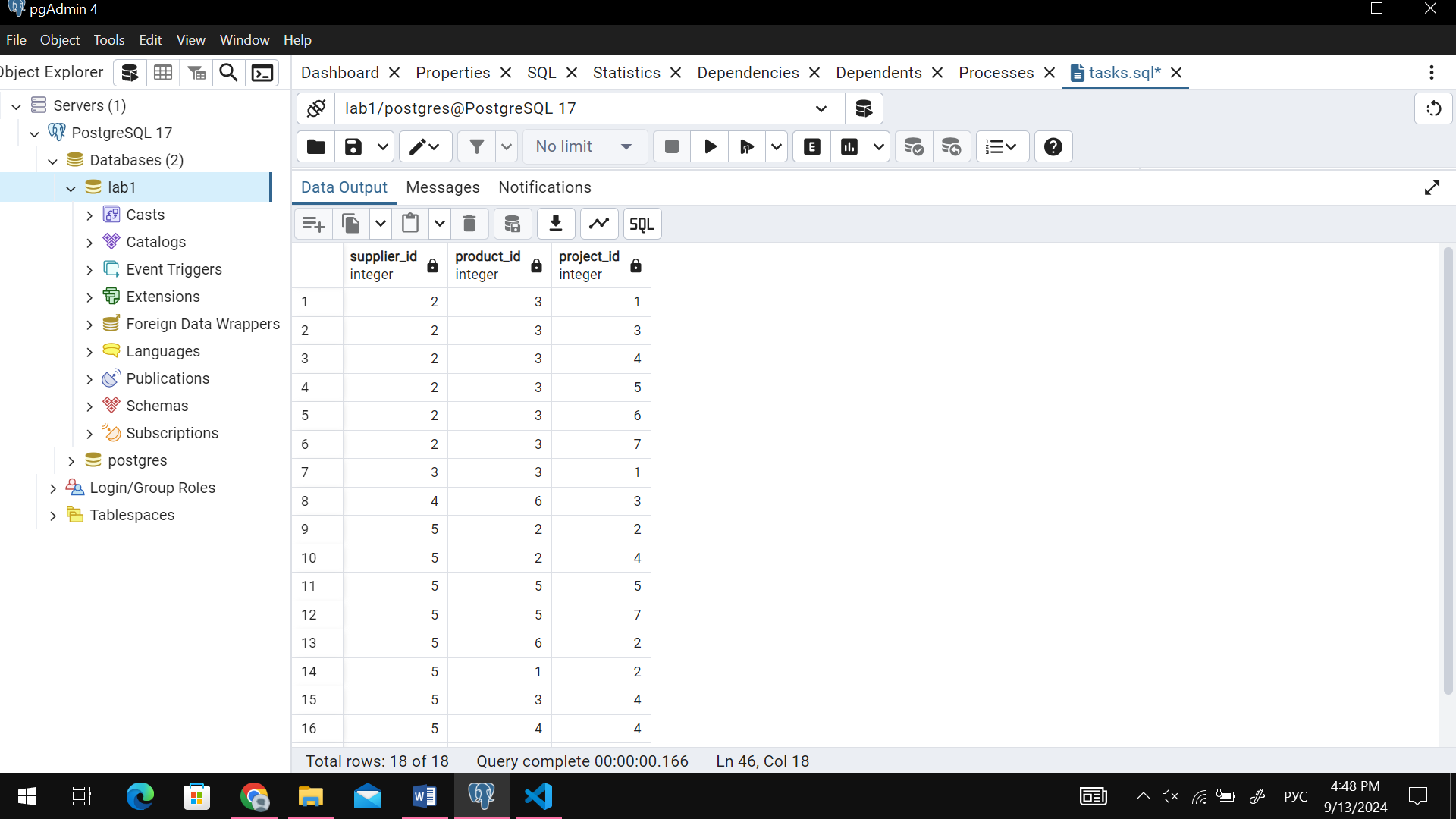
JOIN suppliers s ON s.id = supplier\_id

JOIN products prd ON prd.id = product\_id

JOIN projects prj ON prj.id = project\_id

WHERE s.city != prd.city AND prd.city != prj.city AND s.city != prj.city;

**Результат:**



**Задача №32: Получить номера проектов, обеспечиваемых по крайней мере всеми деталями поставщика П1.**

SELECT q.project\_id

FROM quantities q

WHERE q.product\_id IN

    (SELECT DISTINCT q.product\_id

    FROM quantities q

    WHERE q.supplier\_id = 1)

GROUP BY q.project\_id

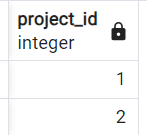
HAVING COUNT(DISTINCT q.product\_id) =

    (SELECT COUNT(DISTINCT q.product\_id)

    FROM quantities q

    WHERE q.supplier\_id = 1);

**Результат:**

****

**Задача №2: Получить полную информацию обо всех проектах в Лондоне.**

SELECT p.id AS project\_id, p.name AS project\_name, p.city AS project\_city,

       s.name AS supplier\_name, s.city AS supplier\_city,

       pr.name AS product\_name, pr.color AS product\_color, pr.size AS product\_size,

       q.quantity

FROM projects p

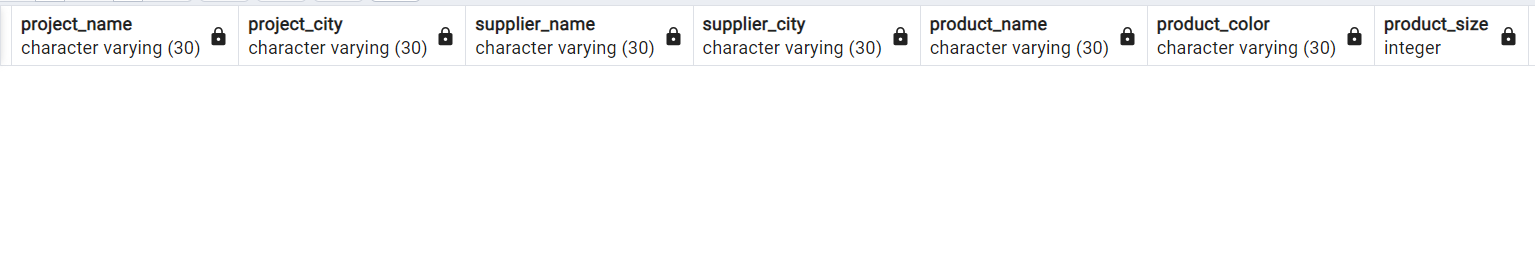
JOIN quantities q ON p.id = q.project\_id

JOIN suppliers s ON q.supplier\_id = s.id

JOIN products pr ON q.product\_id = pr.id

WHERE p.city = 'Лондон';

**Результат:**

****

**Задача №18: Получить номера деталей, поставляемых для некоторого проекта со средним количеством больше 320.**

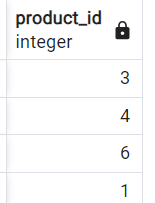
SELECT q.product\_id

FROM quantities q

GROUP BY q.product\_id

HAVING AVG(q.quantity) > 320;

**Результат:**

****