ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

# Бази даних та інформаційні системи

**ЛАБОРАТОРНА РОБОТА №2**

**Транзакції в СКБД PostgreSQL**

Виконав:

Ст. Яцуляк Андрій

Група ПМі-31

Оцінка

Перевірила:

доц. Малець Р.Б.

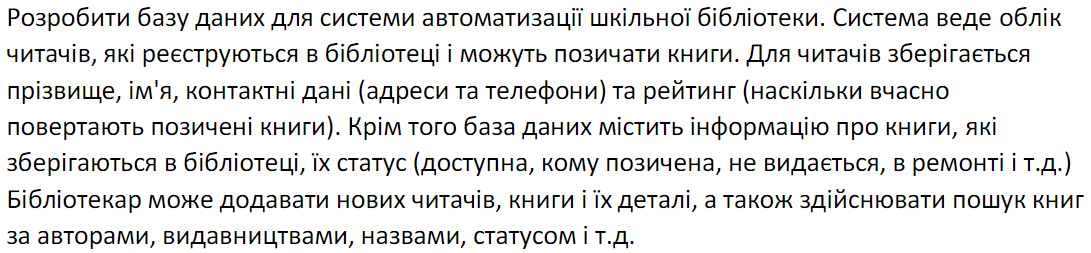
2023

**Тема**: Вивчення понять транзакції та управління конкурентним доступом в СКБД

PostgreSQL.

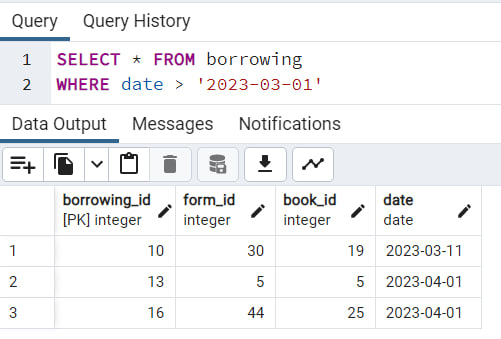
**Мета роботи:** Ознайомлення з використанням транзакцій, їх розробкою та застосуванням, рівнями ізоляцій та механізмом управління конкурентним доступом в СКБД PostgreSQL.

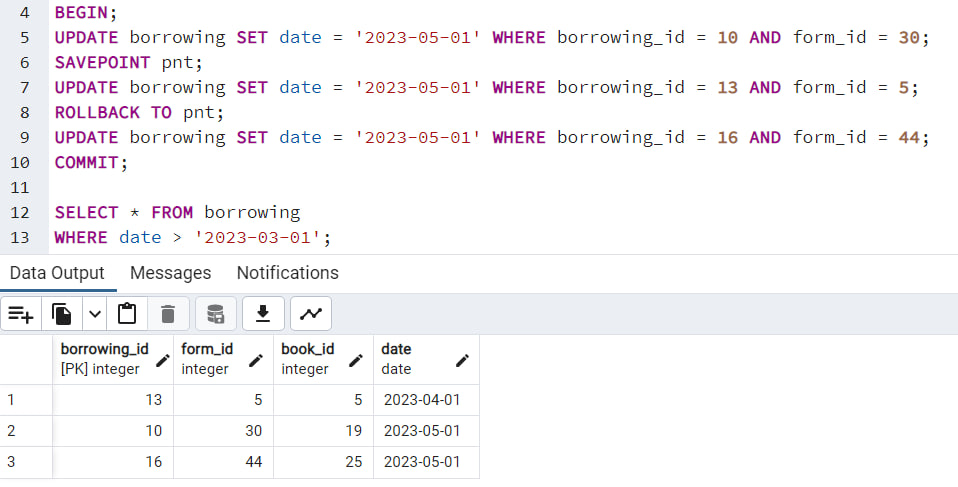
**Завдання**



## Хід роботи

1. Опрацював теоретичний матеріал.
2. Написав першу транзакцію, використовуючи SAVEPOINT та ROLLBACK. Приклад роботи:



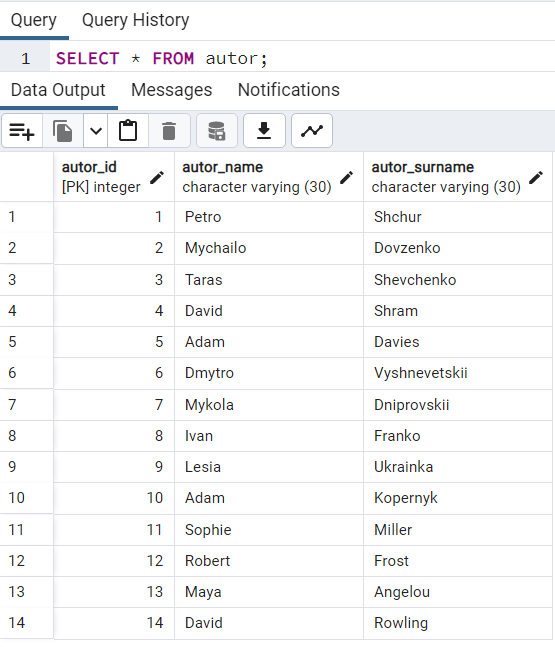


Як ми бачимо, взята книга з borrowing\_id = 13 не змінила дату взяття через ролбек.

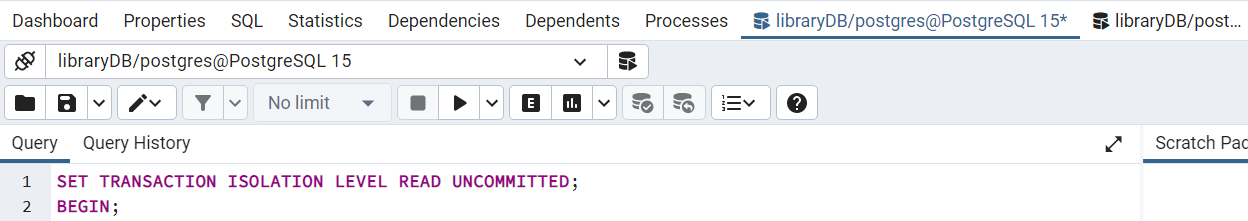
1. Написав транзакції, щоб продемонструвати як вони допомагають вирішити різні проблеми під час паралельних запитів:

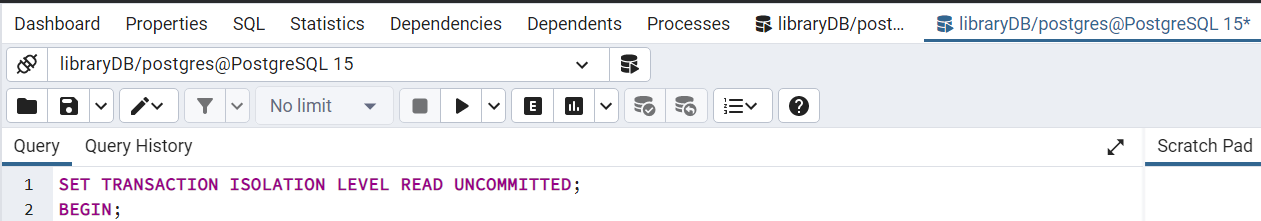
* dirty read - відбувається, коли одна транзакція зчитує дані, які були змінені, але ще не зафіксовані іншою транзакцією:

Покажу це на таблиці AUTOR:

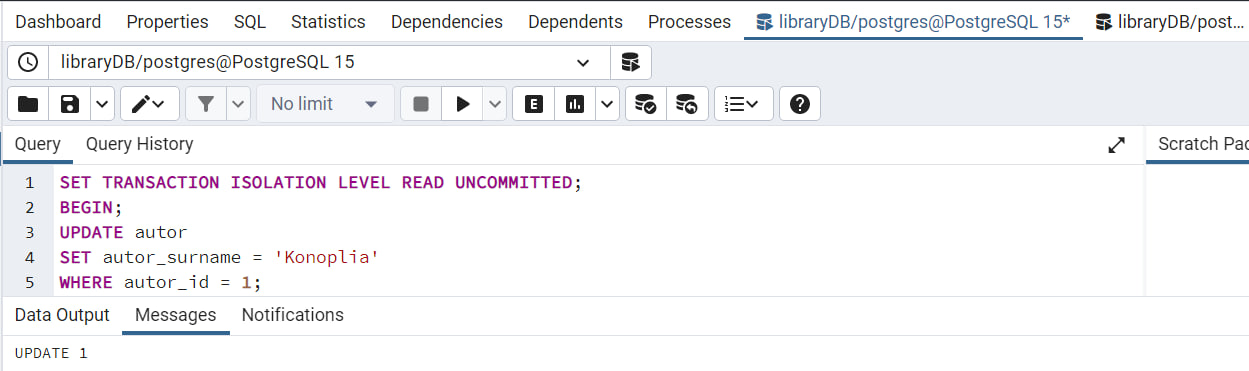


В середовищі pgAdmin для своєї бази даних відкрив дві вкладки Query Tool. В обох вказав рівень ізоляції ‘Read Uncommitted’:

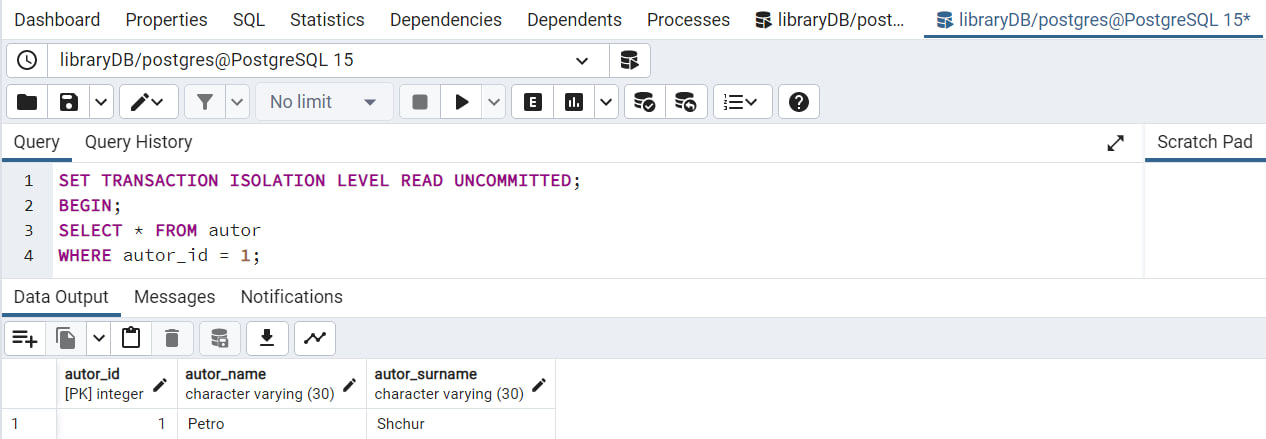




В першій вкладці оновлюю рядок, але не підтверджую

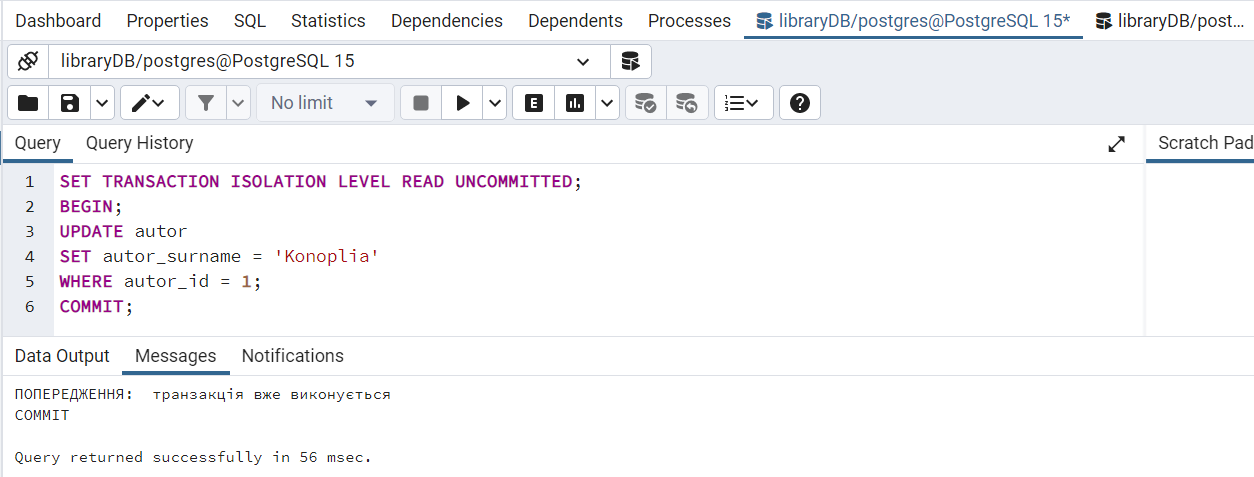


В другій вкладці пишу SELECT вираз

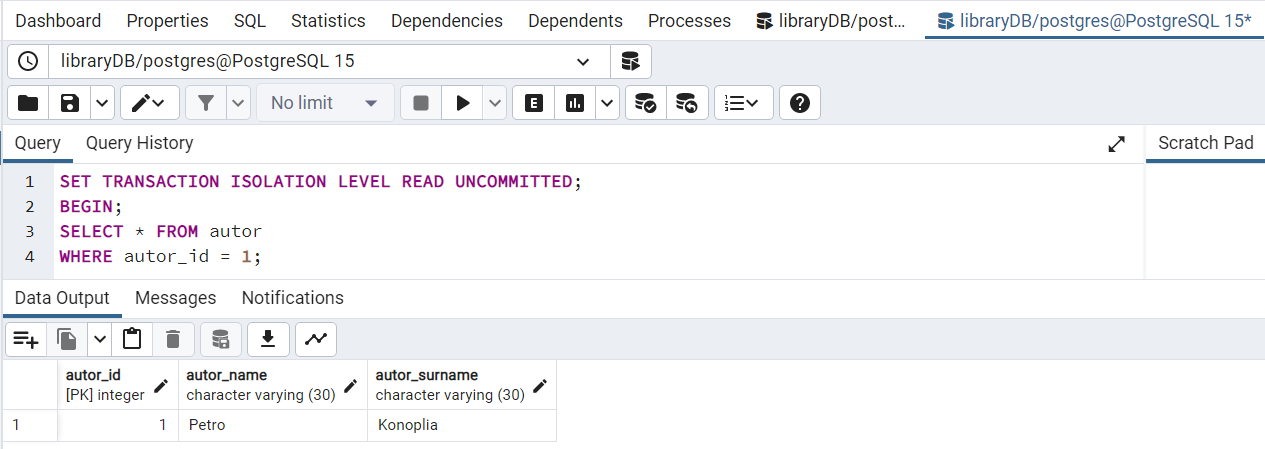


Отже, друга вкладка не помітила незафіксованої зміни. Дописую COMMIT до

першої вкладки



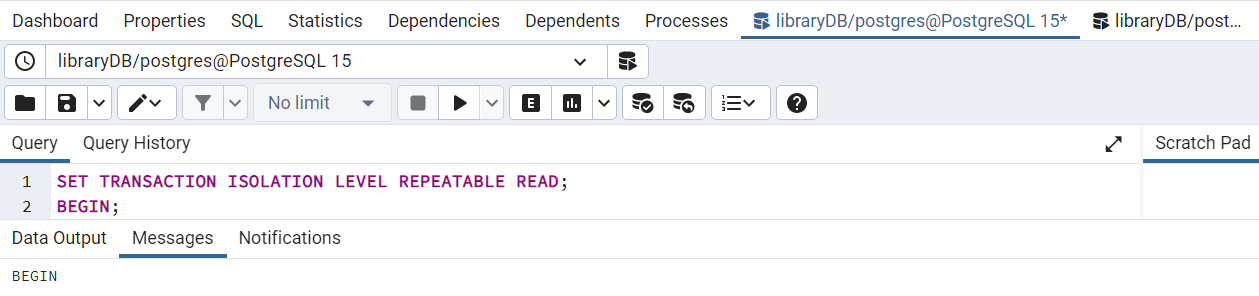
і запускаю аналогічний запит у 2-й

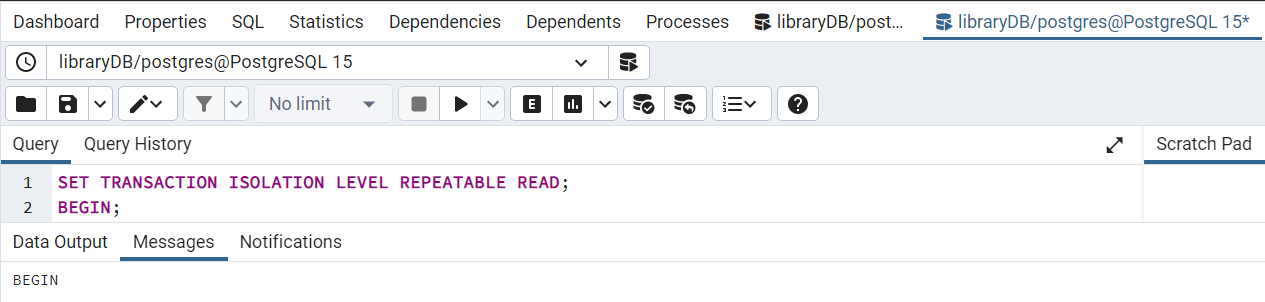


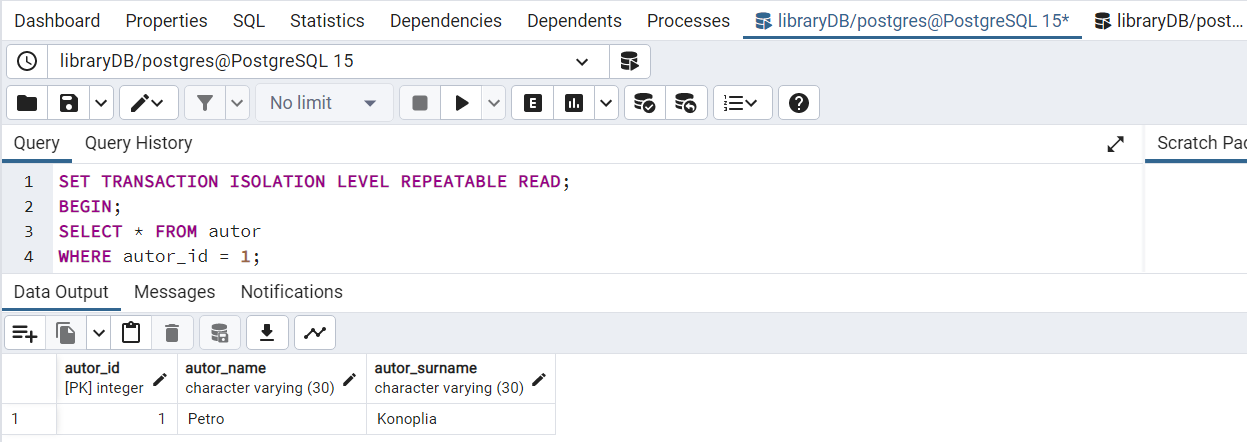
Як ми бачимо, зміни відбулися у другій вкладці

* Nonrepeatable Read - відбувається, коли транзакція зчитує ті самі дані кілька разів, але отримує різні результати, оскільки інша транзакція змінила дані між ними:

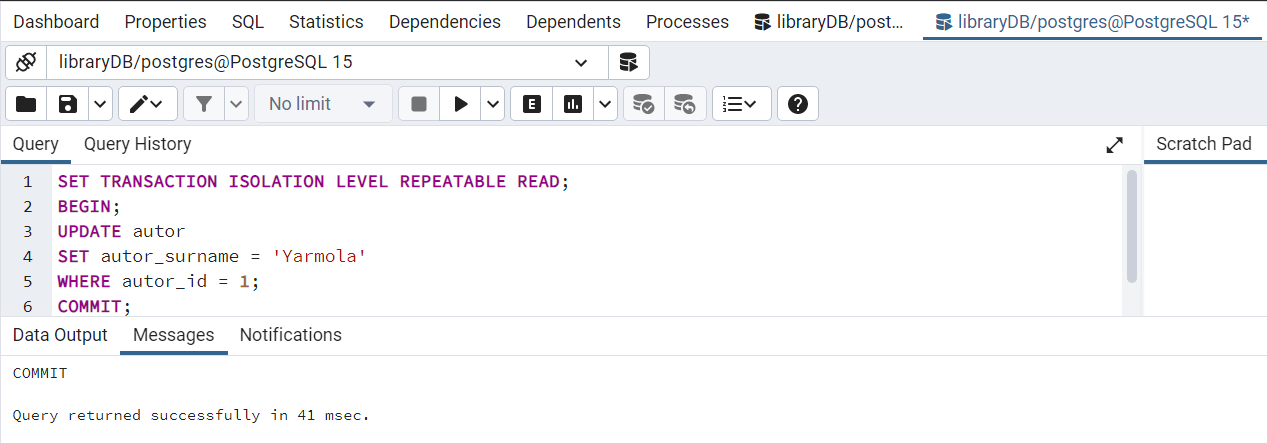
В обох вкладках встановив рівень ізоляції ‘Repeatable read’



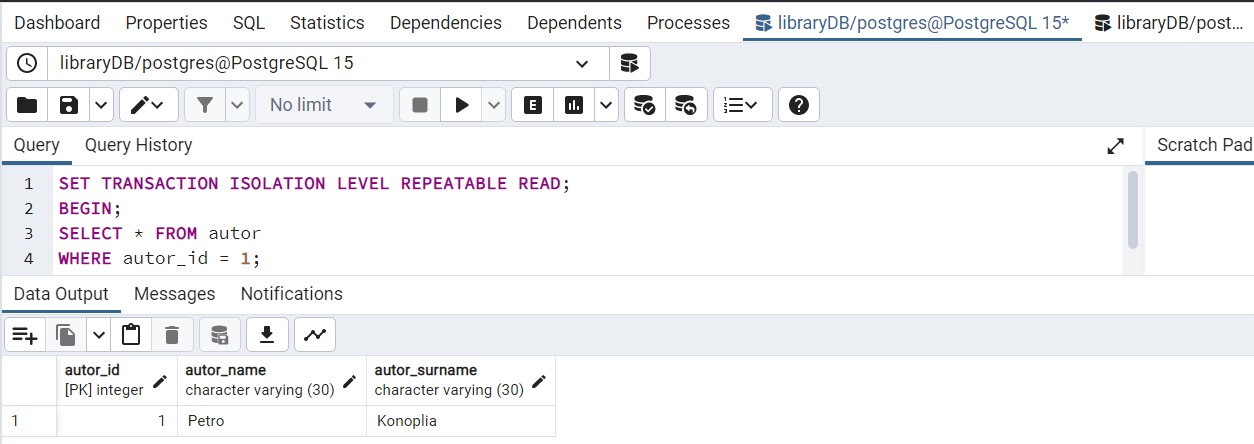


В 1-й вкладці пишу SELECT запит 

В 2-й вкладці оновлюю той самий рядок



В 1-й вкладці виконую той самий SELECT запит

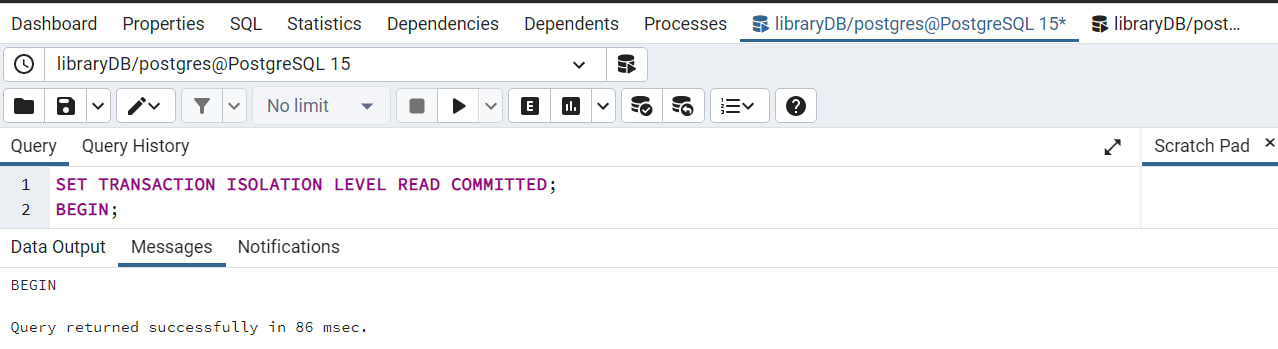


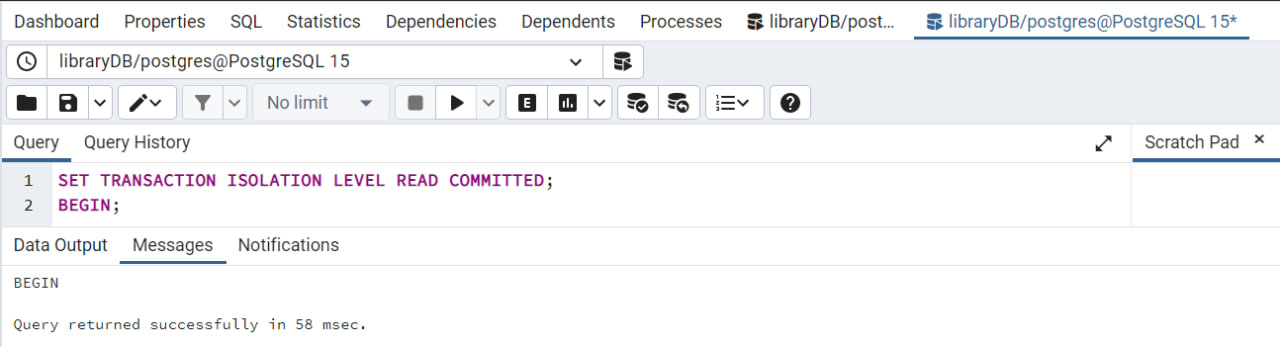
Отже, дані не змінилися. Це демонструє, що ‘Repeatable read’ запобігає ‘Nonrepeatable Read’, надаючи послідовний порядок даних протягом усієї

транзакції, гарантуючи, що та сама операція читання дає той самий результат, навіть якщо інші транзакції змінюють дані.

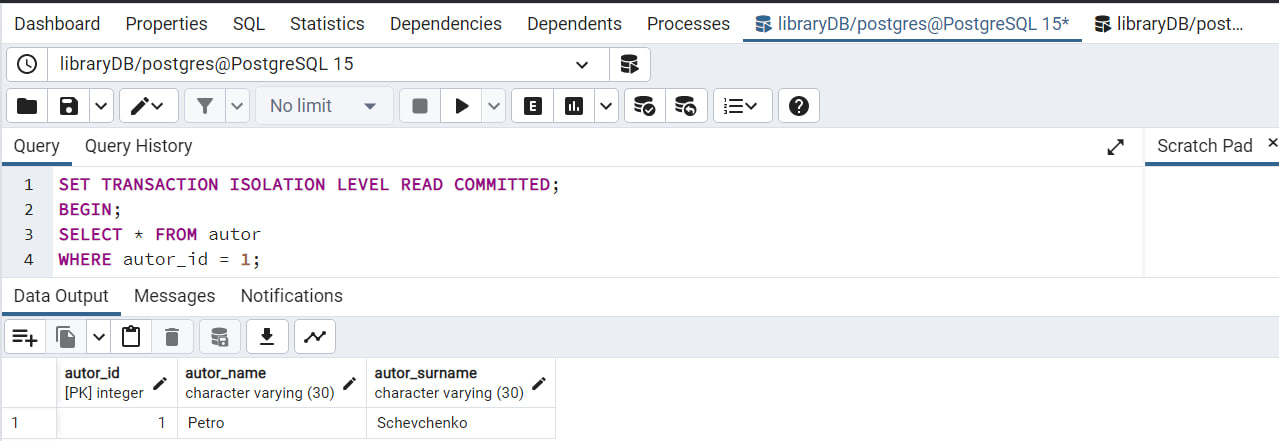
Але я використав рівень ізоляції ‘Read commited’, де продемонстував виконання того самого коду:

В обох вкладках встановив рівень ізоляції ‘Read commited’

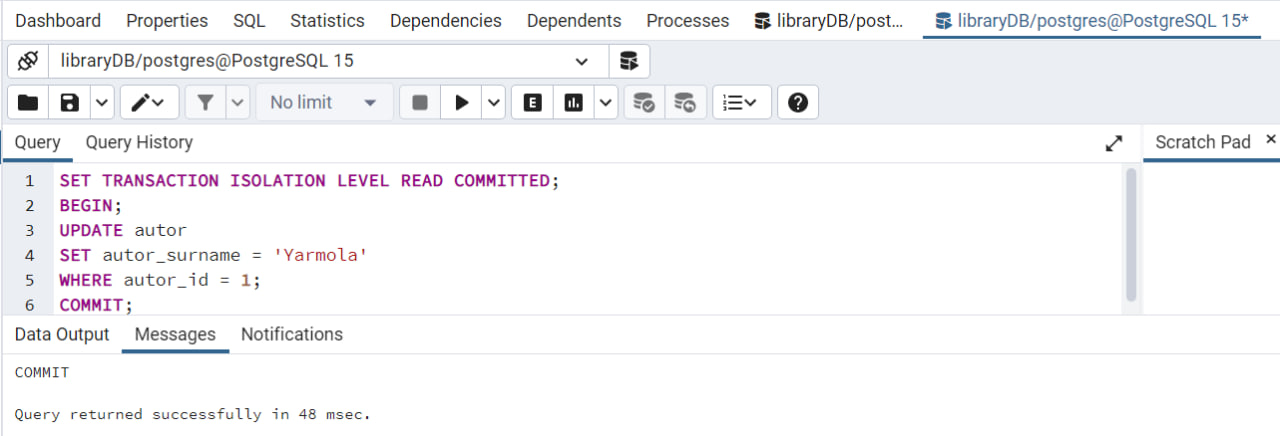




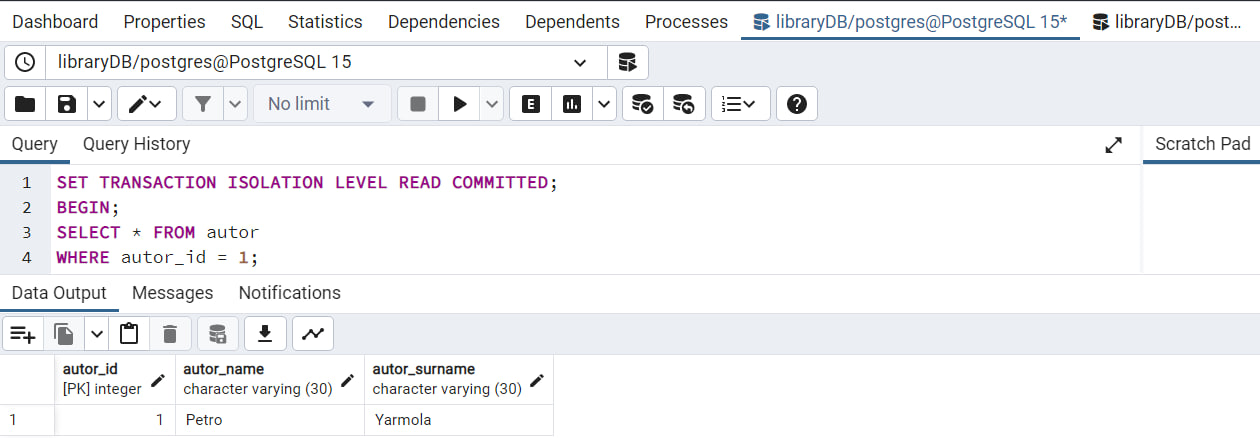
В 1-й вкладці пишу SELECT запит



В 2-й вкладці оновлюю той самий рядок



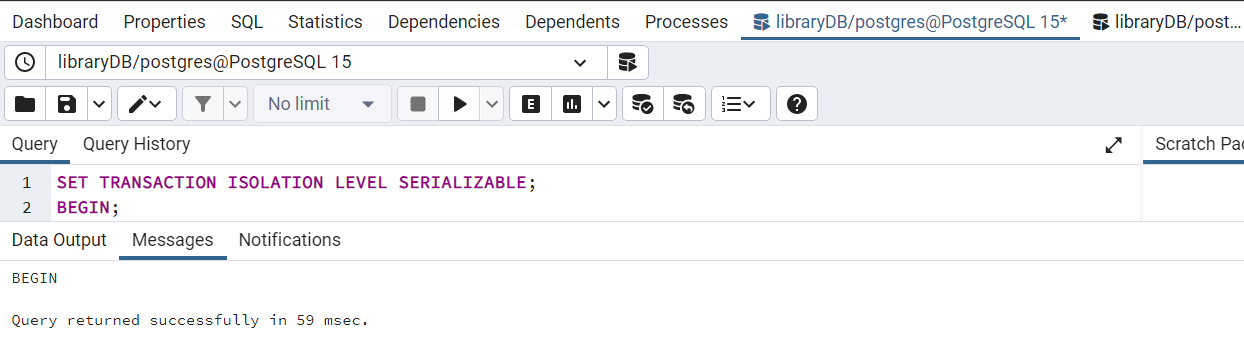
В 1-й вкладці виконую той самий SELECT запит

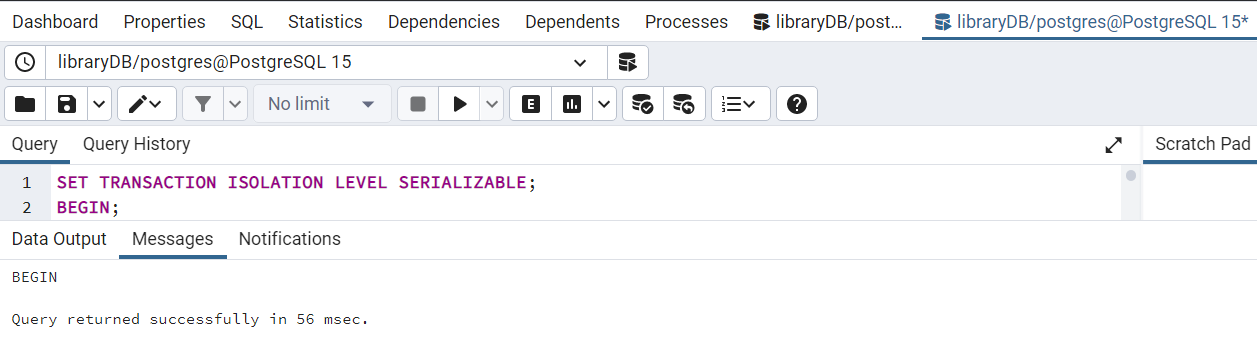


Як бачимо, дані змінились. ‘Read commited’ не запобігає ‘Nonrepeatable Read’.

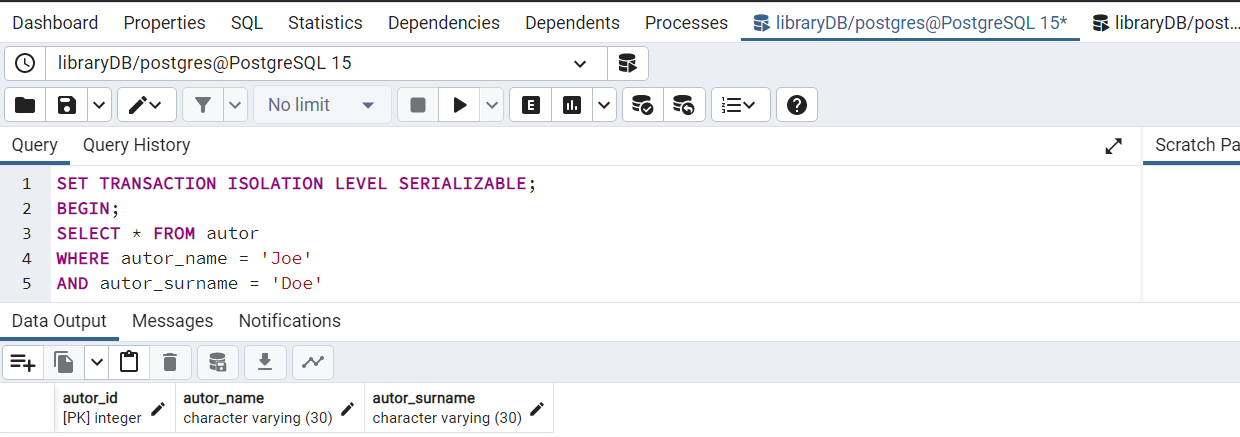
* Phantom read - відбувається, коли транзакція читає набір рядків, які задовольняють умову, але інша транзакція вставляє або видаляє рядки, які також задовольняють цю умову між читаннями.

В обох вкладках встановив рівень ізоляції ‘Serializable’

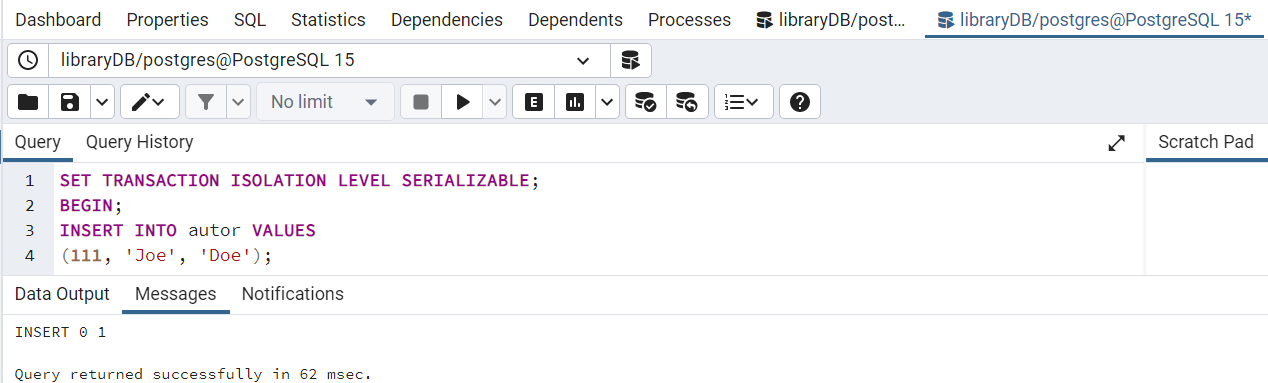




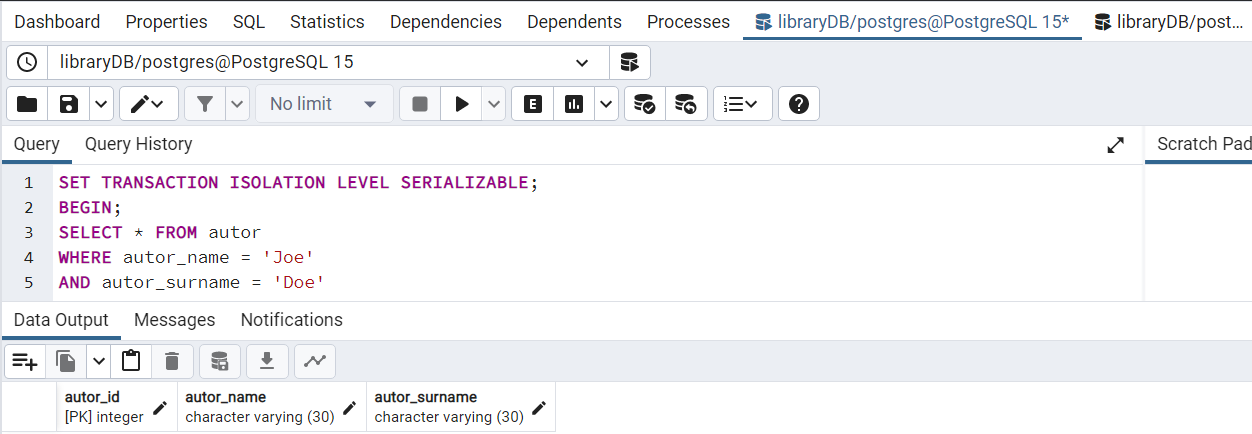
В 1-й вкладці написав SELECT запит



В 2-й вкладці вставив у таблицю новий рядок



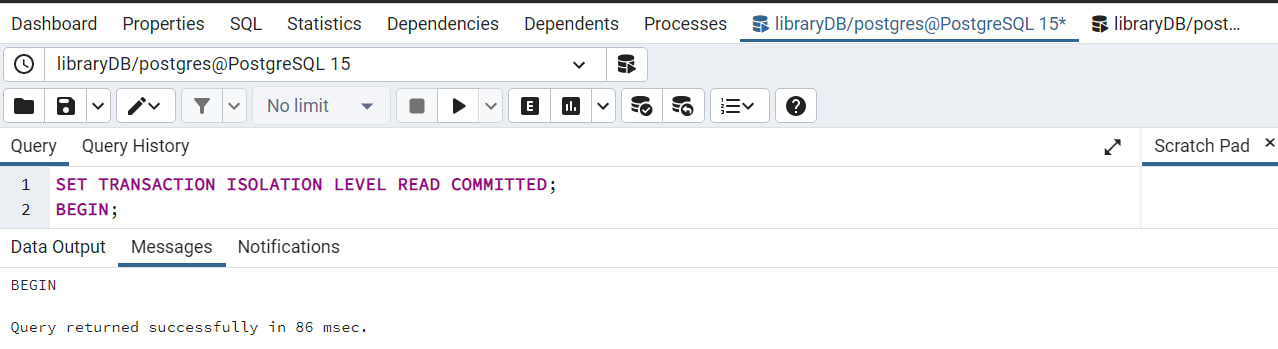
В 1-й вкладці запускаю аналогічний запит SELECT

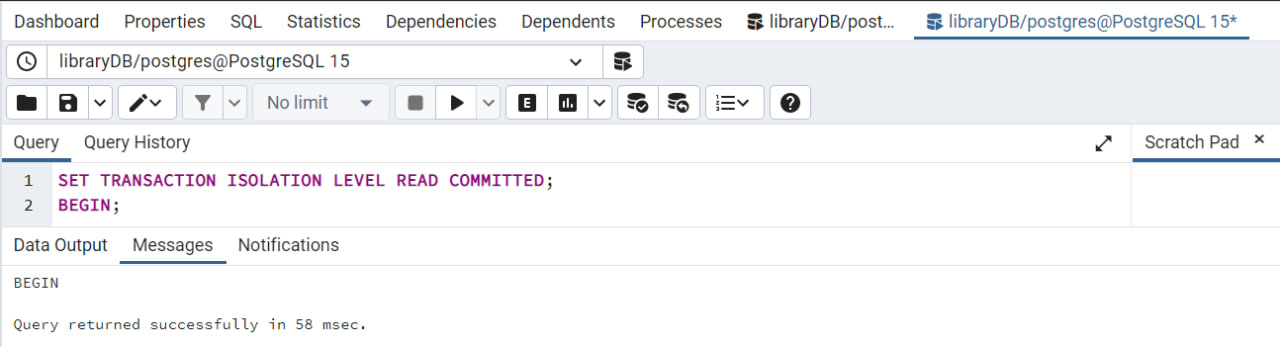


Нічого не змінилося, тому що ‘Serializable’ запобігає явищу ‘Phantom Read’.

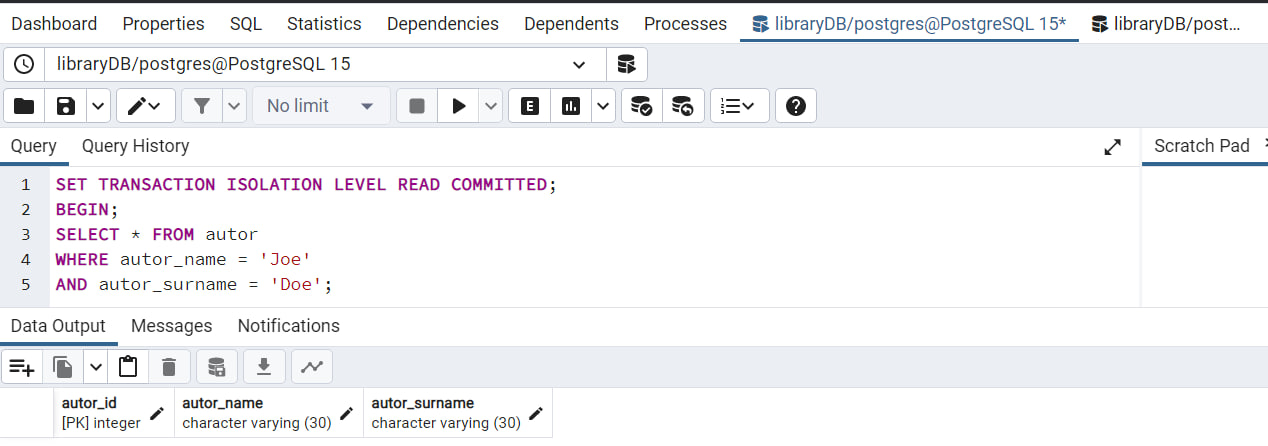
Але я використав рівень ізоляції ‘Read commited’, де продемонстував виконання того самого коду:

В обох вкладках встановив рівень ізоляції ‘Read commited’

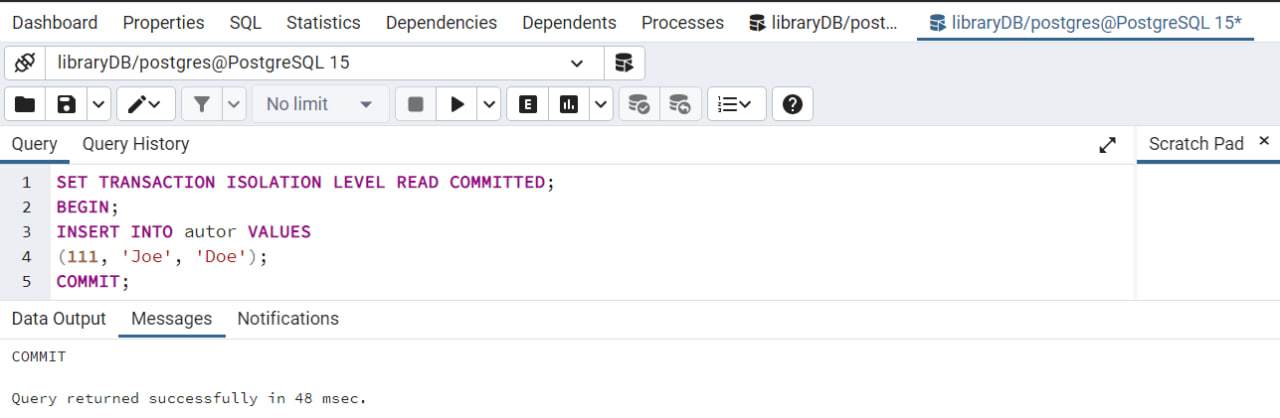




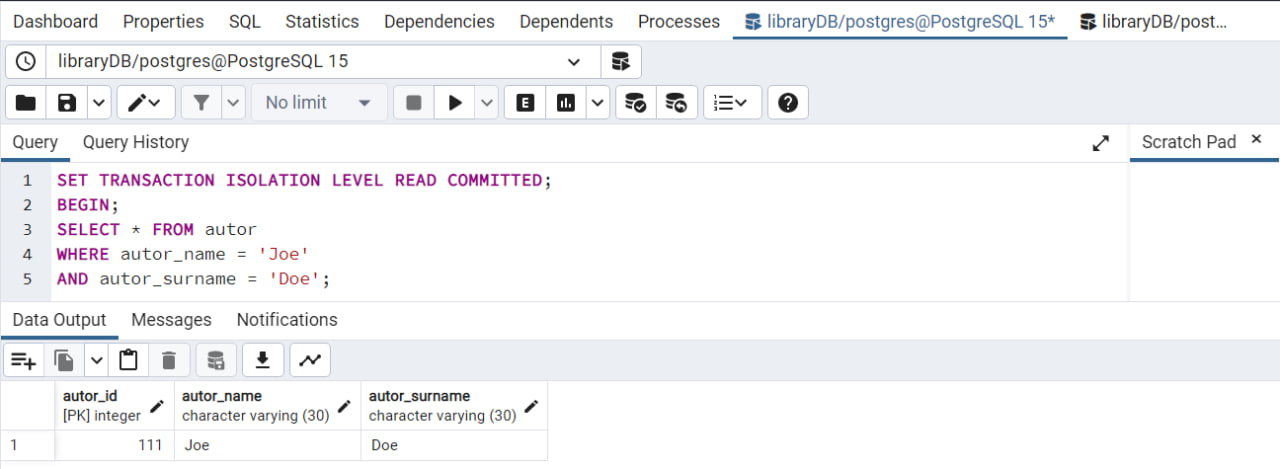
В 1-й вкладці пишу SELECT запит



В 2-й вкладці вставив у таблицю новий рядок



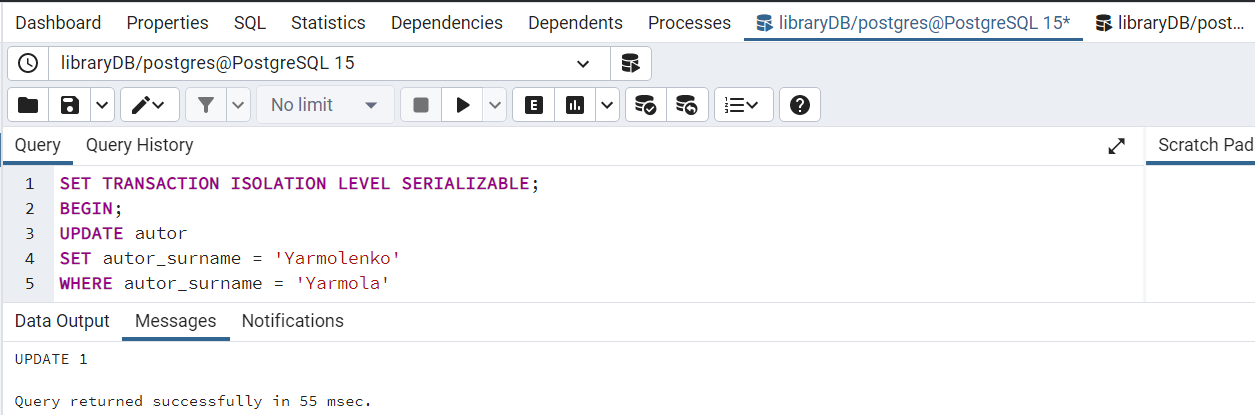
В 1-й вкладці запускаю аналогічний запит SELECT



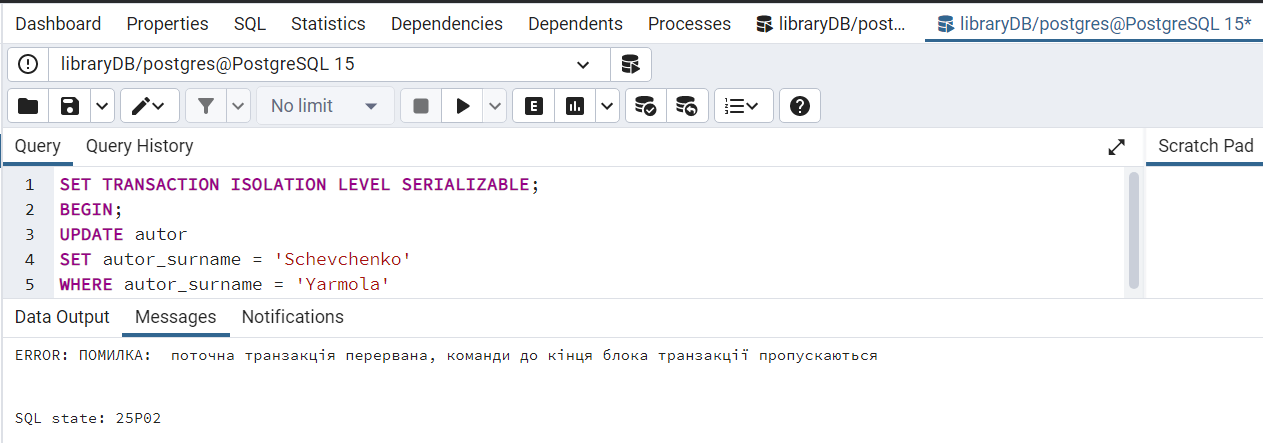
Дані змінились.

* Serialization anomaly - виникає, коли транзакції, які здаються серіалізованими, дають результати, які неможливі під час послідовного виконання тих самих транзакцій:

Оновив рядок в першій вкладці



Написав конфліктуючий запит у 2-й вкладці



Цей запит не виконався, оскільки ‘Serializable’ запобігає явищу ‘Serialization

Anomaly’.

**Висновок:** під час виконання лабораторно роботи я ознайомився з використанням

транзакцій, їх розробкою та застосуванням, рівнями ізоляцій та механізмом

управління конкурентним доступом в СКБД PostgreSQL.