**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ ІНСТИТУТ КОМП’ЮТЕРНИХ СИСТЕМ**

**КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ**

Лабораторна робота №10 з дисципліни

**“** Операційні системи**”**

Тема

«Керування процесами-транзакціями в базах даних.

Частина 2»

Викона

Караульний Д.М

Перевірили: Блажко О.А Дрозд М.О.

# Одеса 2021

**Мета роботи:** дослідити поведінку процесів-транзакцій в базах даних та засоби керуванням ними через механізм блокування з використанням

сучасних систем керування базами даних.

# Хід роботи:

1. Перелік завдань до лабораторної роботи
2. Результатами виконання пунктів завдань
3. Висновки

# Перелік завдань до лабораторної роботи:

**Завдання 1. Аналіз роботи багато версійного протоколу**

Підготуйте чотири транзакції за прикладом з рисунку 2:

− T1 – отримання номеру транзакції, внесення нового рядка в таблицю та перегляд вмісту таблиці;

− T2 – постійний перегляд вмісту таблиці

− T3 – видалення рядку з наступною відміною цієї операції;

− T4 – зміна значення однієї з колонок рядка.

В операцію читання рядка таблиці додайте системні колонки xmin, xmax. На кожному кроці виконання транзакції переглядайте значення колонок xmin, xmax.та зробіть відповідні висновки.

# Завдання 2. Аналіз стану транзакцій на різних рівнях багаторівневого блокування

Виконайте послідовно в двох терміналах наступні комбінації блокувань таблиці:

IX-IS, SIX-IX, SIX-IS. Надайте висновки про сумісність блокувань.

Для кожної комбінації блокувань перед завершенням 1-ї транзакції (яка розпочалася раніше) в додатковому терміналі через команду psql отримайте данні про стан транзакцій (таблиця pg\_locs).

# Завдання 3. Керування квазіпаралельним виконанням транзакцій на різних рівнях ізоляції транзакцій

Підготуйте транзакції, які було створено у завданні 3.1 рішення попередньої

лабораторної роботи, а саме, створіть дві транзакції, кожна з яких повинна включати такі операції:

* операція читання першого рядку таблиці;
* операція редагування однієї із змінних таблиці в першому рядку;
* повторна операція читання першого рядку таблиці;
* операція фіксації всіх змін.
  1. Виконайте роботу транзакцій при умові їх роботи на рівні ізоляції READ COMMITED. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї

транзакції (яка виконується пізніше) та дайте свої висновки.

* 1. Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції

REPEATABLE READ. Проаналізуйте реакцію СКБД на операцію UPDATE 2 ї транзакції (яка виконується пізніше) та дайте свої висновки.

* 1. Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції SERIALIZABLE. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

# Завдання 4. Керування квазіпаралельним виконанням транзакцій при наявності тупикових ситуацій.

* 1. Виконайте модифікацію транзакцій так, щоб вони призводили до тупикової ситуації.
  2. Виконайте дві модифіковані транзакції. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та яка призвела до тупику. Дайте свої висновки з урахуванням:

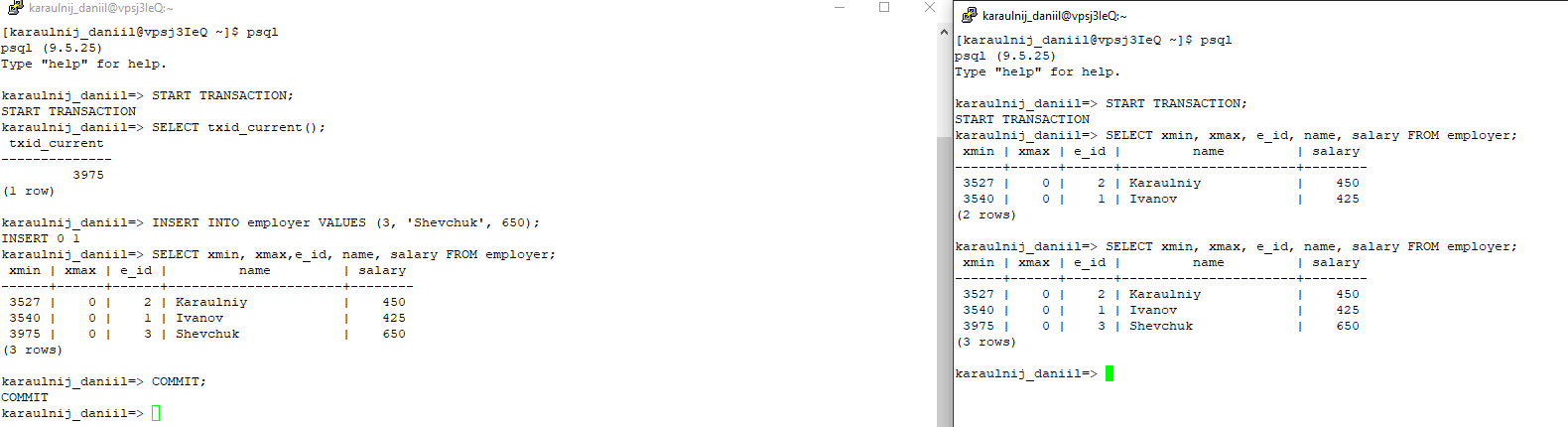
− ідентифікаторів процесів

− номерів транзакцій.

# Результати виконання завдань:

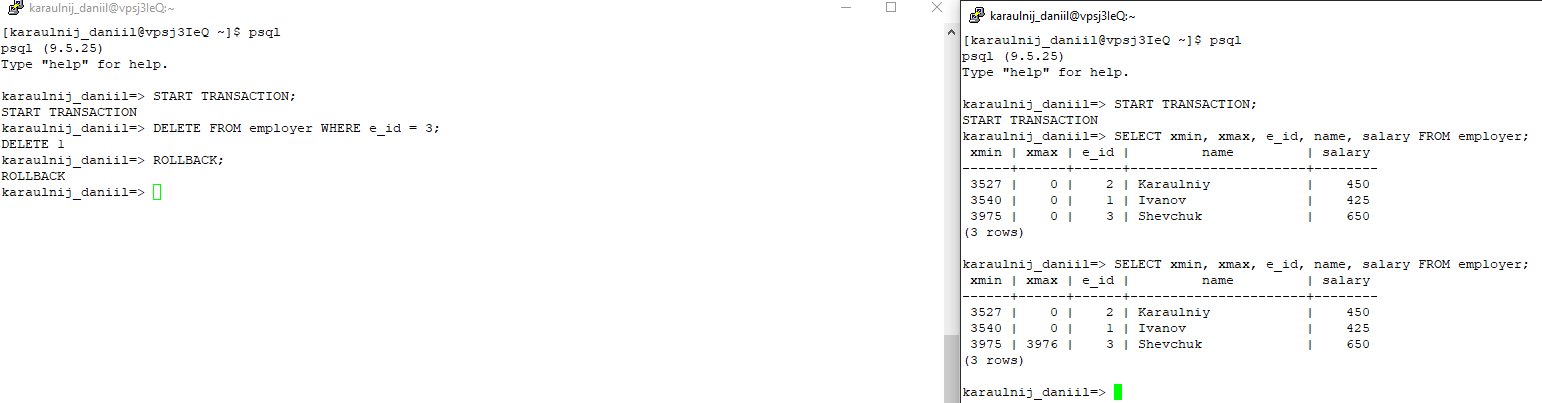
**Завдання 1. Аналіз роботи багато версійного протоколу**

*Транзакції 1 та 2:*



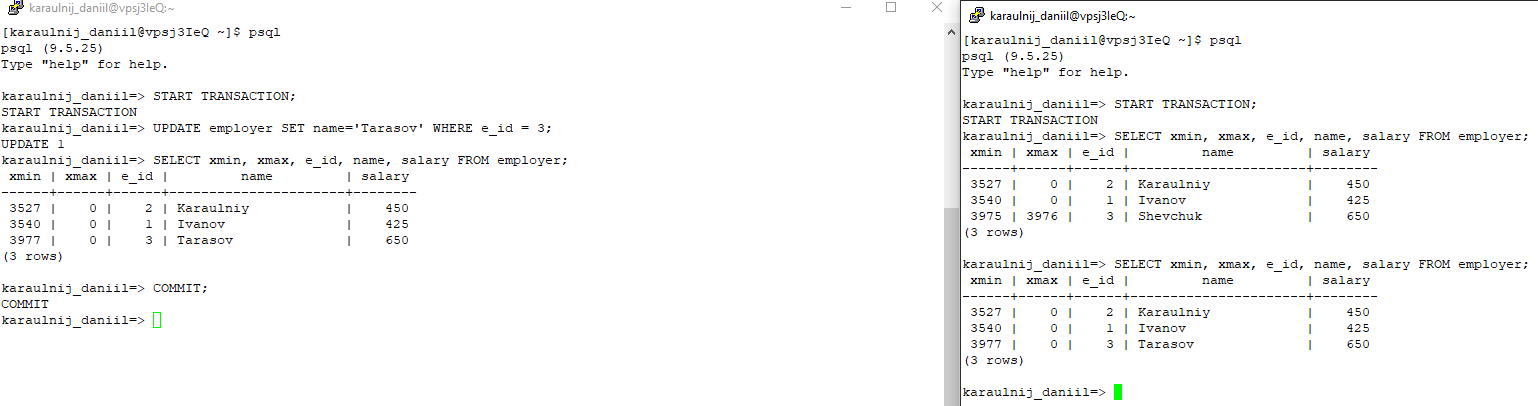
*Поки зміни у 1 транзакції не зафіксовано, транзакція 2 не реагує на зміни, виконані в 1 транзакції. Після фіксації змін у 1 транзакції, 2 транзакція може обробляти змінену інформацію. Як підсумок, ми бачимо, що у таблицю додається новий рядок, з xmin = 3975, xmax = 0.*

*Транзакції 3 та 2:*



*Ми бачимо, що після видалення 3 рядку та відміни операції xmax рядку набуває значення 3976. Це означає, що над 3 рядком здійснювалася операція під номером 3976.*

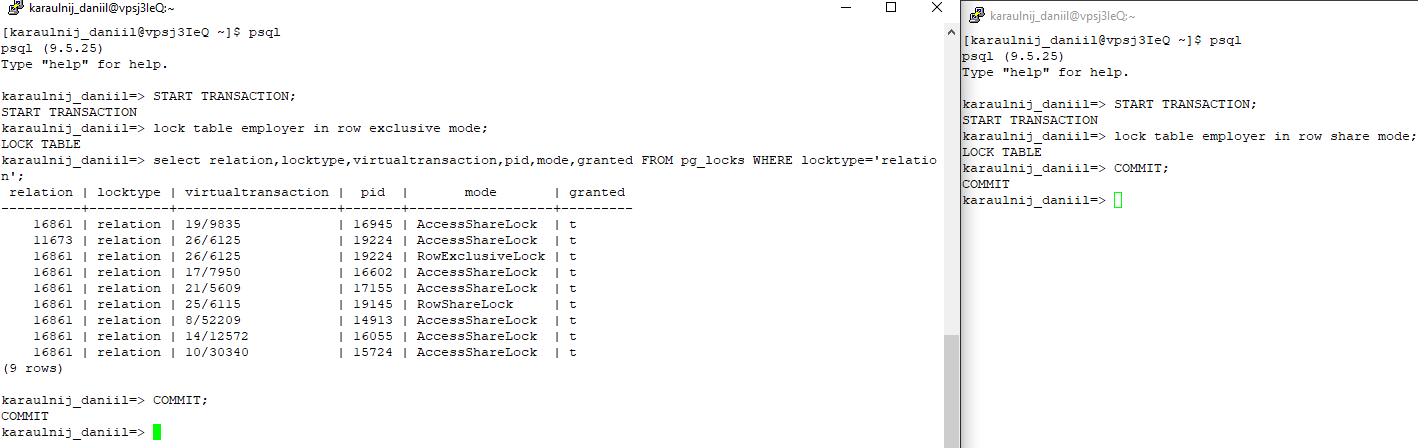
*Транзакції 4 та 2:*



*Значення xmax 3рядку, над яким виконувалась операція, змінилось на 3977, після фіксації цих змін xmin перейняло значення 3977 , а xmax = 0, яке показує, що поточне значення було створено транзакцією з номером 3977, але поки немає нових версії, створених іншими транзакціями.*

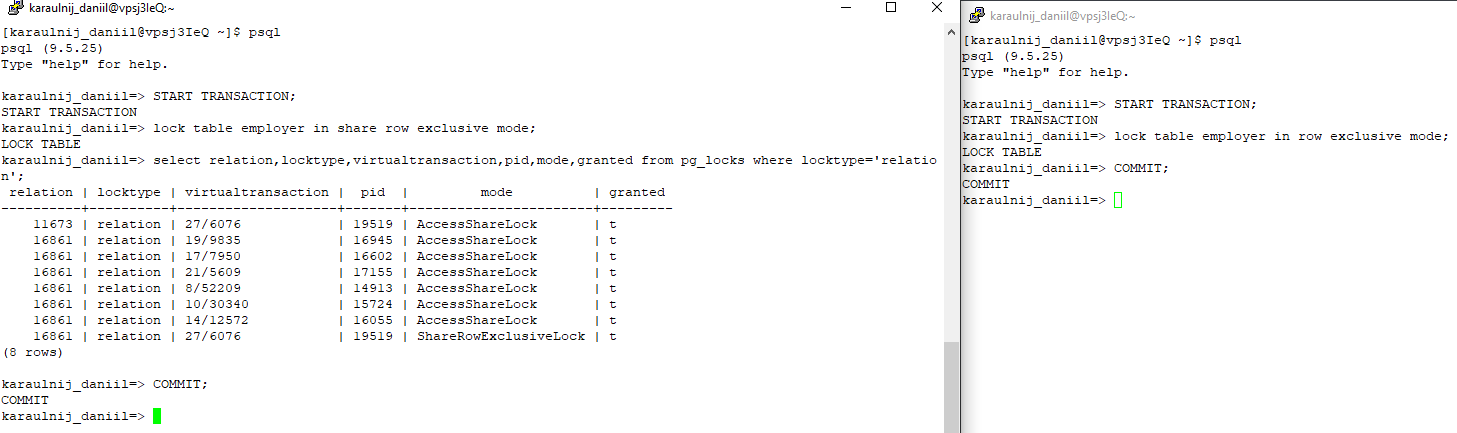
# Завдання 2. Аналіз стану транзакцій на різних рівнях багаторівневого блокування

*Комбінації блокувань IX-IS:*



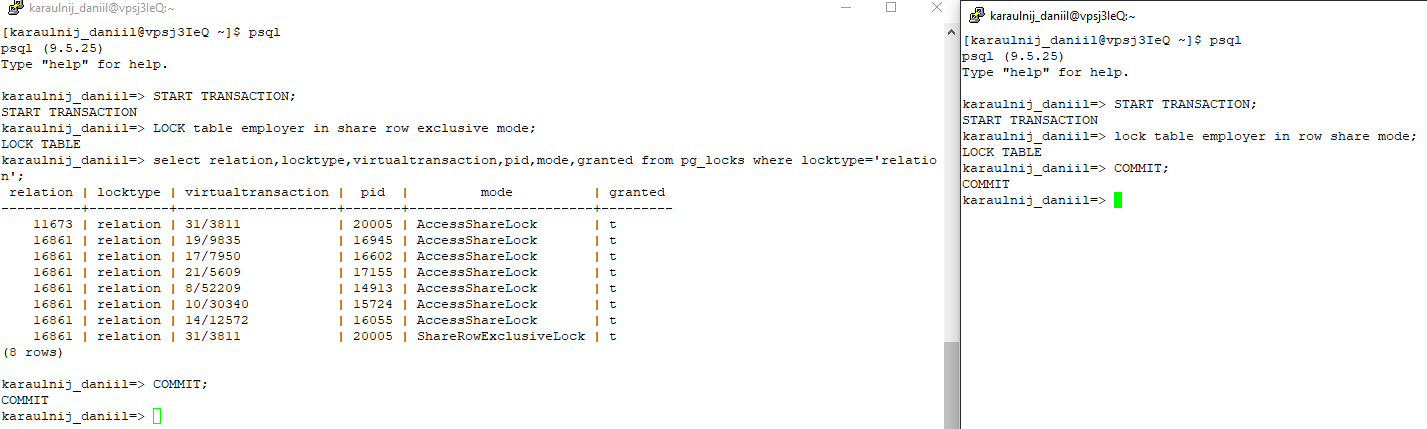
*Блокування IX та IS сумісні. Підтвердження цьому можна знати в таблиці pg\_locs, оскільки стан блокування для процесу 19145 дорівнює t - операцію блокування виконано;*

*Комбінації блокувань SIX-IS:*



*Блокування SIX та IX не сумісні. Підтвердження цьому можна знати в таблиці pg\_locs, оскільки стан блокування для процесу 19519 дорівнює f - операція блокування чекає через несумісність з іншими вже виконаними операціями блокування.*

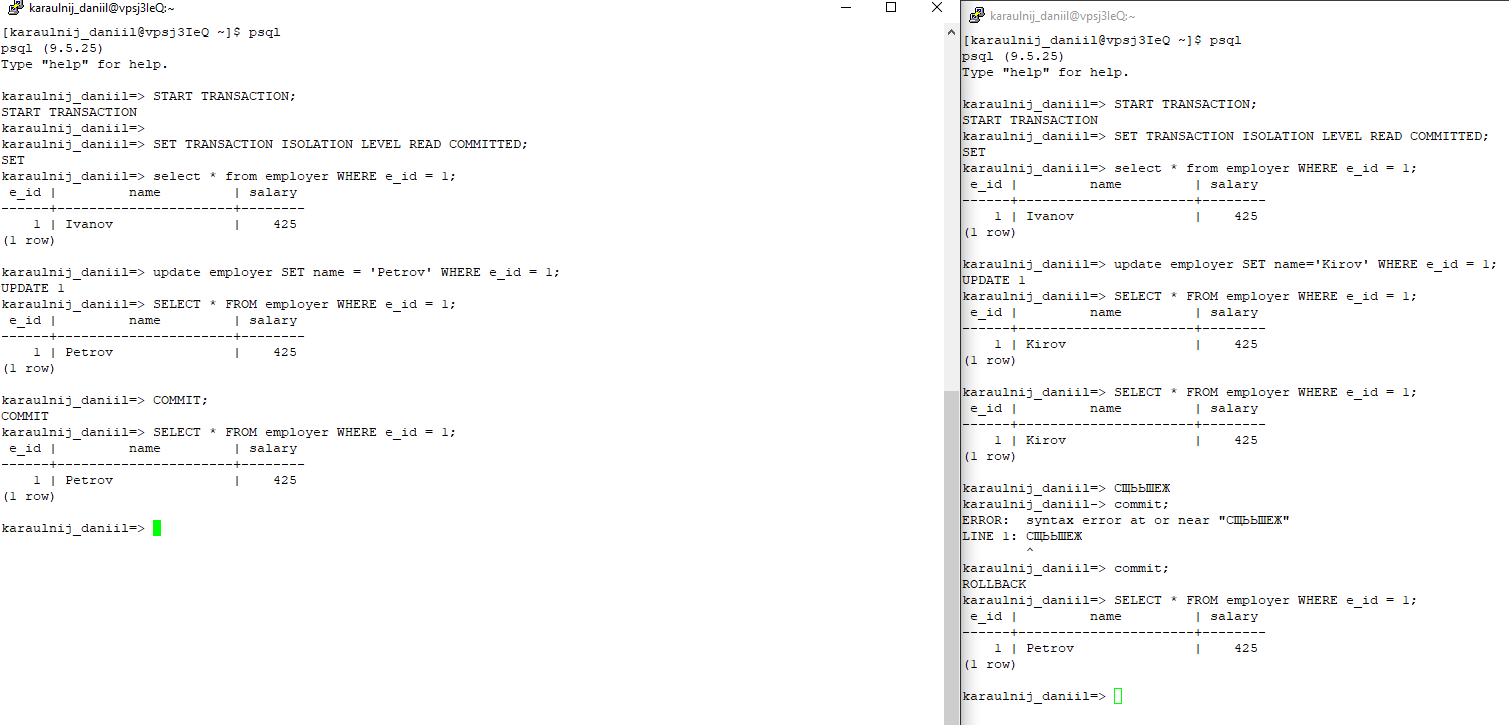
*Комбінації блокувань SIX-IS:*



*Блокування SIX та IS сумісні. Підтвердження цьому можна знати в таблиці pg\_locs, оскільки стан блокування для процесу 20005 дорівнює t - операцію блокування виконано;*

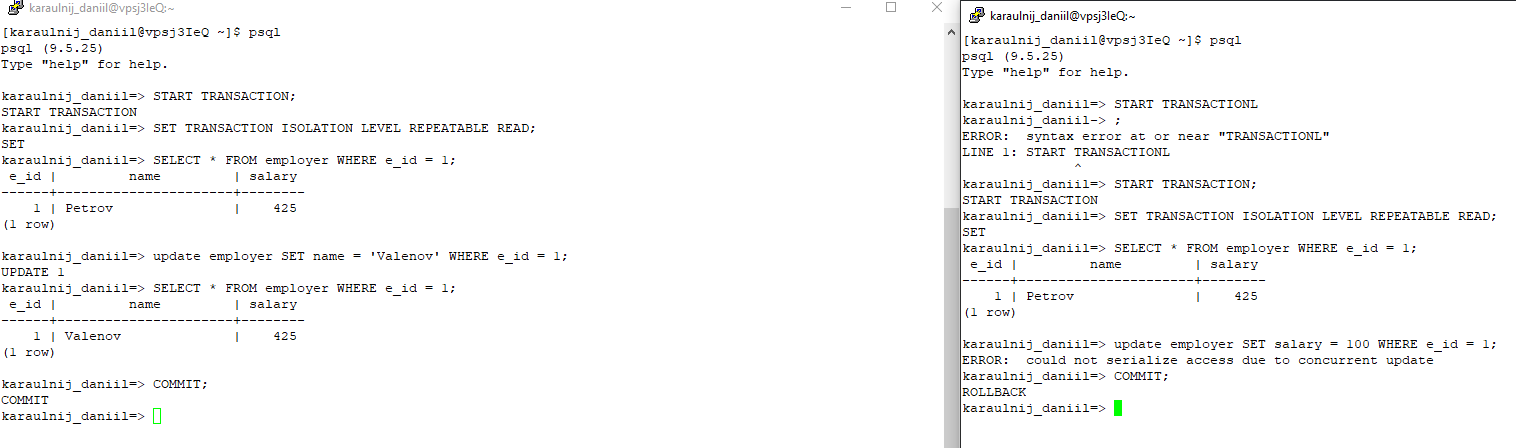
# Завдання 3. Керування квазіпаралельним виконанням транзакцій на різних рівнях ізоляції транзакцій

*Рівень ізоляції READ COMMITED:*



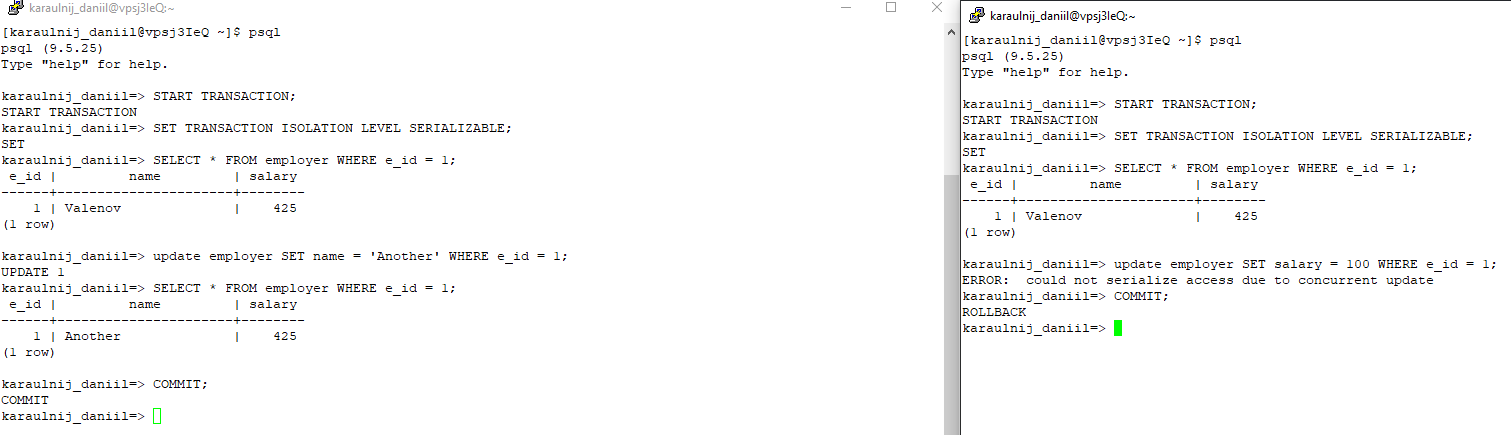
*Виконуючи операцію UPDATE 2-ї транзакції, транзакція переходить в режим очікування.. Тільки після того, як 1-ша транзакція виконає операцію COMMIT, 2-га транзакція побачить зміни*

*Рівнень ізоляції REPEATABLE READ:*



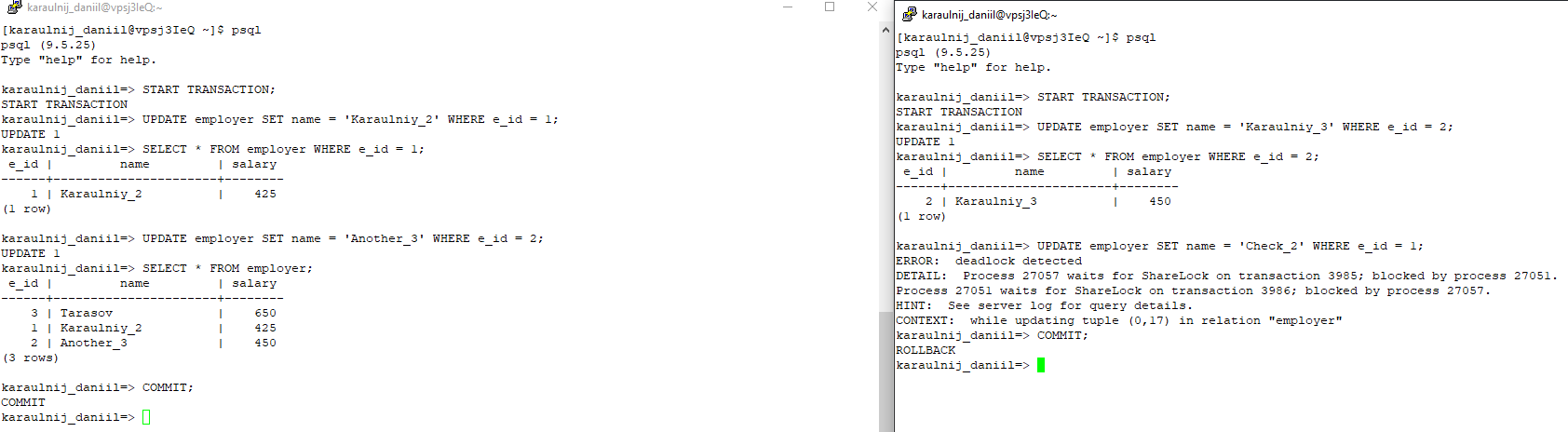
*Виконуючи операцію UPDATE 2-ї транзакції, транзакція переходить в режим очікування, після завершення 1-ї транзакції повідомляє про помилку та завершує транзакцію без зміни даних. Це через те, що 2-га транзакція не побачить зміни, яка виконала 1-ша транзакція навіть після виконання операції COMMIT.*

*Рівнень ізоляції SERIALIZABLE:*



*Можемо помітити, що після виконання операції UPDATE у 2-ій транзакції, транзакцію буде переведено в режим очікування, а вже після завершення 1-ї транзакції, ми отримаємо повідомлення про помилку, при цьому Т2 буде завершена без змін. Це через те, що на такому рівні ізоляції заборонено виконувати паралельно зміни одних даних.*

# Завдання 4. Керування квазіпаралельним виконанням транзакцій при наявності тупикових ситуацій.

*Виконаємо модифікацію транзакцій так, щоб вони призводили до тупикової ситуації.*

*При виконанні операції UPDATE 2-ї транзакції, було отримано повідомлення про помилку, через виникнення тупикової ситуації. СКБД примусово скасувала 2-гу транзакцію, команда якої привела до тупикової ситуації, а 1- шу транзакція зберегла.*

**Висновок:** Під час виконання лабораторної роботи було досліджено поведінку процесів-транзакцій в базах даних та засоби керуванням ними через механізм блокування з використанням сучасних систем керування базами даних. Найскладнішим було завдання з аналізу роботи багато версійного протоколу.