**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**“ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**



**Лабораторна робота № 11**

з дисципліни

«Організація баз даних та знань»

з теми :

**“**Розробка та застосування транзакцій**”**

**Виконав:**

Студент групи КН-209

Музика Роман

**Перевірила:**

К.т.н. Мельникова Н. І.

Львів – 2020р

**Тема роботи**

Розробка та застосування транзакцій.

**Мета роботи**

Навчитися використовувати механізм транзакцій у СУБД MySQL. Розробити SQL запити, які виконуються як єдине ціле в рамках однієї транзакції.

**Короткі теоретичні відомості**

Транзакція – це сукупність директив SQL, які виконуються як єдине ціле з можливістю відміни результатів їх виконання. Зміни в таблицях записуються у базу даних лише після успішного виконання всіх директив транзакції. Інакше, всі зроблені зміни ігноруються. Це дозволяє уникати помилок при маніпулюванні великими обсягами записів, зберігати цілісність даних при помилках під час додавання, видалення, модифікації значень у різних таблицях і полях тощо. СУБД MySQL також підтримує глобальні розподілені транзакції, які виконуються на декількох базах даних, або на різних серверах баз даних (ХА-транзакції).

Для організації транзакцій в MySQL використовують такі директиви, як *SET autocommit, START TRANSACTION, COMMIT і ROLLBACK*.

*START TRANSACTION* Вказує на початок транзакції. Директива вимикає автоматичне збереження змін для всіх подальших запитів, поки не буде виконано команду *COMMIT*, або *ROLLBACK*.

*COMMIT* Зберегти зміни, зроблені даною транзакцією.

*ROLLBACK* Відмінити дану транзакцію і зроблені нею зміни у базі даних. Слід зауважити, що зміни у схемі бази даних не можна відмінити, тобто результат видалення, зміни або створення таблиці завжди зберігається.

*SET autocommit=0* Вимикає автоматичне збереження змін для поточної сесії зв’язку з сервером БД. За замовчуванням, зміни зберігаються автоматично, тобто результат виконання запиту, який змінює таблицю, одразу записується на диск без можливості відміни операції.

*AND CHAIN* Одразу після завершення даної транзакції розпочати виконання наступної.

*RELEASE* Одразу після виконання даної транзакції завершити поточну сесію зв’язку з сервером.

Транзакції можна розбивати на окремі логічні частини, оголошуючи так звані точки збереження. Це дозволяє відміняти результати виконання не всієї транзакції, а лише тих запитів, які виконувались після оголошеної точки збереження *(SAVEPOINT).*

*SAVEPOINT* мітка Оголошує точку збереження всередині транзакції та задає її назву.

*ROLLBACK TO [SAVEPOINT]* мітка Відміняє результати виконання запитів, вказаних після даної точки збереження.

*RELEASE SAVEPOINT* мітка Видаляє точку збереження.

**Хід роботи**

Розробимо транзакцію, яка буде вносити дані в таблицю Event з подальшим її виведенням. Після чого ми повертаємося до початкового вигляду нашої БД.

Транзакція складається з трьох запитів на додавання нових подій у таблицю Event та запиту на її виведення.



Висновок

У даній лабораторній роботі ми ознайомилися із механізмом транзакцій у СУБД MySQL.