

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”
ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра систем штучного інтелекту

Лабораторна робота №11
із дисципліни
Бази даних

Виконав:
Ст. групи КН-207
Гуменчук Б.Р.
Прийняв:
Мельникова Н.І.

Львів – 2018 р.

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 (XA-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 відмінити 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.

1. Невдала транзакція.

```
174 • start transaction;
175 • insert into performer values (4, "Volodymir", "2000-10-20", "volodyamort@gmail.com");
176 • insert into performer values (5, "Sviatoslav", "1991-04-04", "sviatik_bos@gmail.com");
177 • commit;
```

Output			
Action Output			
#	Time	Action	Message
775	22:52:51	SET SQL_MODE=@OLD_SQL_MODE	0 row(s) affected, 1 warning(s): 3135 'NO_ZERO_DATE'.
776	22:52:51	SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS	0 row(s) affected
777	22:52:51	SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS	0 row(s) affected
778	22:52:51	select * from performer LIMIT 0, 1000	3 row(s) returned
779	22:52:51	start transaction	0 row(s) affected
780	22:52:51	insert into performer values (4, "Volodymir", "2000-10-20", "volodyamort@gmail.com")	Error Code: 1062. Duplicate entry '4' for key 'PRIMARY'

2. Вдала транзакція

```
174 • start transaction;
175 • insert into performer values (5, "Volodymir", "2000-10-20", "volodyamort@gmail.com");
176 • insert into performer values (6, "Sviatoslav", "1991-04-04", "sviatik_bos@gmail.com");
177 • commit;
```

Output			
Action Output			
#	Time	Action	Message
796	22:54:38	SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS	0 row(s) affected
797	22:54:38	select * from performer LIMIT 0, 1000	3 row(s) returned
798	22:54:38	start transaction	0 row(s) affected
799	22:54:38	insert into performer values (5, "Volodymir", "2000-10-20", "volodyamort@gmail.com")	1 row(s) affected
800	22:54:38	insert into performer values (6, "Sviatoslav", "1991-04-04", "sviatik_bos@gmail.com")	1 row(s) affected
801	22:54:38	commit	0 row(s) affected

Висновок: на цій лабораторній роботі було розглянуто транзакції в MySQL.