

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІНСТИТУТ КОМП'ЮТЕРНИХ СИСТЕМ
КАФЕДРА «ІНФОРМАЦІЙНІ СИСТЕМИ»

Лабораторна робота №10

З дисципліни

«Операційні системи»

Тема

«Керування процесами-транзакціями в базах даних»

Виконав:

Студент групи АІ-203

Вояковський Д. П.

Перевірили:

Дрозд М.О. Блажко О.А.

Одеса 2021

Завдання для виконання

Для кожної транзакції підготуйте окремий термінал, в якому виконайте команду доступу до вашої БД з використанням утиліти `psql`.

Завдання 1 Аналіз роботи багатоверсійного протоколу

В завданні 1 рішення попередньої лабораторної роботи було створено таблицю з декількома рядками.

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_locks`).

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.

1.1 Виконайте роботу транзакцій при умові їх роботи на рівні ізоляції READ COMMITTED. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

1.2 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.

1.3 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.

3.1 Виконайте модифікацію транзакцій так, щоб вони призводили до тупикової ситуації.

3.2 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.

Завдання 1

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> insert into student values (19,'Sidorov',3);
INSERT 0 1
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2827 |    0 |   19 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> delete from student where name = 'Sidorov';
DELETE 1
voyakovskij_dmitro=> rollback;
ROLLBACK
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 17 where name = 'Sidorov';
UPDATE 1
voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=> █
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> insert into student values (64,'Fedorov',2);
INSERT 0 1
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2830 |    0 |   64 | Fedorov   |    2
(4 rows)

voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> delete from student where name = 'Fedorov';
DELETE 1
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> rollback;
ROLLBACK
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 79 where name = 'Fedorov';
UPDATE 1
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2832 |    0 |   79 | Fedorov   |    2
(4 rows)

voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=> █
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
(2 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2827 |    0 |   19 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2827 | 2828 |   19 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2827 | 2829 |   19 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
(3 rows)
```

```
^ voyakovskij_dmitro=> start transaction;
WARNING: there is already a transaction in progress
START TRANSACTION
voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
(3 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2830 |    0 |   64 | Fedorov   |    2
(4 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2830 | 2831 |   64 | Fedorov   |    2
(4 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2830 | 2832 |   64 | Fedorov   |    2
(4 rows)

voyakovskij_dmitro=> select xmin,xmax,s_id,name,kurs from student;
  xmin | xmax | s_id | name      | kurs
-----+-----+-----+-----+-----
 2150 |    0 |    3 | Ivanov    |    2
 2215 | 2218 |   30 | Petrov    |    5
 2829 |    0 |   17 | Sidorov   |    3
 2832 |    0 |   79 | Fedorov   |    2
(4 rows)
```

Завдання 2

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> lock table student in row exclusive mode;
LOCK TABLE

voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> lock table student in share row exclusive mode;
LOCK TABLE

voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> lock table student in share row exclusive mode;
LOCK TABLE

voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> lock table student in row share mode;
LOCK TABLE
```

```
voyakovskij_dmitro=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_locks where locktype = 'relation';
 relation | locktype | virtualtransaction | pid | mode | granted
-----+-----+-----+-----+-----+-----
 11673 | relation | 4/85213 | 26831 | AccessShareLock | t
 16642 | relation | 2/720097 | 20436 | RowShareLock | t
 16642 | relation | 3/82050 | 20445 | RowExclusiveLock | t
(3 rows)
```

```
voyakovskij_dmitro=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_locks where locktype = 'relation';
 relation | locktype | virtualtransaction | pid | mode | granted
-----+-----+-----+-----+-----+-----
 11673 | relation | 4/85214 | 26831 | AccessShareLock | t
 16642 | relation | 2/720098 | 20436 | RowExclusiveLock | f
 16642 | relation | 3/82051 | 20445 | ShareRowExclusiveLock | t
(3 rows)
```

```
voyakovskij_dmitro=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_locks where locktype = 'relation';
 relation | locktype | virtualtransaction | pid | mode | granted
-----+-----+-----+-----+-----+-----
 11673 | relation | 4/85215 | 26831 | AccessShareLock | t
 16642 | relation | 2/720099 | 20436 | RowShareLock | t
 16642 | relation | 3/82052 | 20445 | ShareRowExclusiveLock | t
(3 rows)
```

Завдання 3

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> set transaction isolation level read committed;
SET
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
 s_id | name | kurs
-----+-----+-----
 21 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> update student set s_id = 11 where name = 'Ivanov';
UPDATE 1
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
 s_id | name | kurs
-----+-----+-----
 11 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> commit;
```

```
voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=>
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> set transaction isolation level read committed;
SET
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
 s_id | name | kurs
-----+-----+-----
 21 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> update student set s_id = 11 where name = 'Ivanov';
select * from student where name = 'Ivanov';
commit;

voyakovskij_dmitro=> select * from student where name = 'Ivanov';
 s_id | name | kurs
-----+-----+-----
 11 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> commit;
COMMIT
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> set transaction isolation level repeatable read;
SET
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
s_id | name | kurs
-----+-----+-----
11 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> update student set s_id = 21 where name = 'Ivanov';
UPDATE 1
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
s_id | name | kurs
-----+-----+-----
21 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> commit;
```

```
voyakovskij_dmitro=> commit; ERROR: could not serialize access due to concurrent update
COMMIT
voyakovskij_dmitro=> ERROR: current transaction is aborted, commands ignored until end of transaction block
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> set transaction isolation level serializable;
SET
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
s_id | name | kurs
-----+-----+-----
21 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> update student set s_id = 1 where name = 'Ivanov';
UPDATE 1
voyakovskij_dmitro=> select * from student where name = 'Ivanov';
s_id | name | kurs
-----+-----+-----
1 | Ivanov | 1
(1 row)

voyakovskij_dmitro=> commit;
```

```
voyakovskij_dmitro=> commit; commit; ERROR: could not serialize access due to concurrent update
COMMIT
voyakovskij_dmitro=> ERROR: current transaction is aborted, commands ignored until end of transaction block
```

Завдання 4

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Petrov';
UPDATE 1
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Ivanov';
UPDATE 1
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Ivanov';
UPDATE 1
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Ivanov';
UPDATE 1
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Petrov';
UPDATE 1
voyakovskij_dmitro=> update student set s_id = 5 where name = 'Petrov';
UPDATE 1
ERROR: deadlock detected
DETAIL: Process 23007 waits for ShareLock on transaction 2970; blocked by process 21312.
Process 21312 waits for ShareLock on transaction 2971; blocked by process 23007.
HINT: See server log for query details.
CONTEXT: while updating tuple (0,2) in relation "student"
```

Висновки: під час виконання лабораторної роботи усі завдання були помірно складні.