

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ “ОДЕСЬКА ПОЛІТЕХНІКА”

Інститут комп'ютерних систем
Кафедра інформаційних систем

Лабораторна робота №9

з дисципліни “Операційні системи”.

Тема: «Керування процесами-транзакціями в базах даних. Частина 1».

Виконала:
студентка групи АІ-204
Томчук Вікторія Борисівна

Перевірив:
Блажко О.А.

Одеса-2021

Мета роботи: дослідити поведінку процесів-транзакцій в базах даних та засоби керування ними через механізм блокування з використанням сучасних систем керування базами даних.

2 Завдання

Завдання 1 Проектування транзакцій

Нехай задані три транзакції, приклади яких представлено в таблиці 15.

У відповідності з вашим варіантом виконайте наступні теоретичні завдання.

1.1 Створіть історії квазіпаралельного виконання транзакцій для протоколу 1-го ступеня блокування з описом таблиці блокування транзакцій.

1.2 Повторіть попереднє завдання з використанням протоколу 2-го ступеня блокування.

1.3 Для створених історій у 1-му та 2-му пунктів завдань визначте наявність тупика транзакції, створивши граф очікування транзакцій.

No команди	No учасника	Приклади транзакцій
9	1	T1= W[A] R[B] W[A] C1 T2= R[A] W[A] C2 T3= R[D] R[B] W[D] C3

Завдання 2 Налаштування бази даних

Нехай існує база даних, назва якої співпадає з іменем вашого користувача в ОС Linux.

Приклад команди створення реляційної таблиці в базі даних наведено в таблиці 16.

Також в таблиці 1 наведено приклад команди внесення одного рядка в реляційну таблицю бази даних.

1.1 Встановіть з'єднання з вашою базою даних.

1.2 У відповідності із варіантом з таблиці 1 створіть реляційну таблицю в базі даних.

1.3 У відповідності із варіантом з таблиці 1 додайте рядок в реляційну таблицю, створену у попередньому пункті завдання.

1.4 Створіть ще одну операцію внесення рядка в таблицю, який буде відрізнятися значеннями всіх змінних від прикладу з варіанту.

1.5 Перевіряють зміст таблиці.

No команди	No учасника	Операція створення таблиці в базі даних	Операції внесення даних в базі даних
9	1	Create table employer (e_id integer, name char(20), salary integer);	Insert into employer values (1, 'Ivanov', 200);

Завдання 3. Керування квазіпаралельним виконанням транзакцій з використанням команд блокування

1 Створіть дві транзакції, кожна з яких повинна включати такі операції:

- операція читання першого рядку таблиці;
- операція зміни однієї із змінних таблиці в першому рядку;
- повторна операція читання першого рядку таблиці;
- операція фіксації всіх змін.

2 При створенні транзакцій включіть відповідні операції блокування для протоколу 1-го ступеня блокування.

3 У двох терміналах виконайте операції транзакцій при їх квазіпаралельному режимі роботи за умови, що одна з транзакція стартує першою.

4 Повторіть роботу транзакцій, але в першій транзакції замість операції фіксації виконайте операцію відміни.

5 Повторіть пункти 3 та 4 але з використанням протоколу 2-го ступеня блокування.

Завдання 1.1. Створіть історії квазіпаралельного виконання транзакцій для протоколу 1-го ступеня блокування з описом таблиці блокування транзакцій.

$H_{T1T2T3} = X1[A], W1[A], R2[A], R3[D],$
 $R1[B], X2[A] - \text{wait}, R3[B],$
 $X1[A], W1[A], X3[D], W3[D],$
 $U1, C1, X2[A], W2[A], U3, C3, U2, C2$

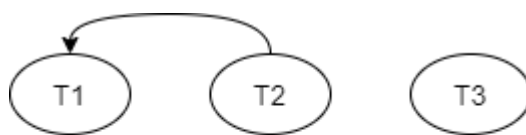
Таблиця блокувань		
Змінна	Блокування	Запити
A	+	X1
	wait	X2
		U1
	+	X2
		U2
B		-
		-
D	+	X3
		U3

Завдання 1.2. Повторіть попереднє завдання з використанням протоколу 2-го ступеня блокування.

$H_{T1T2T3} = X1[A], W1[A], S2[A] - \text{wait}, S3[D], R3[D],$
 $S1[B], R1[B], S2[A] - \text{wait}, S3[B], R3[B],$
 $X1[A], W1[A], S2[A] - \text{wait}, S3[D], R3[D],$
 $U1, C1, S2[A], R2[A], U3, C3, X2[A], W2[A], U2, C2$

Таблиця блокувань		
Змінна	Блокування	Запити
A	+	X1
	wait	S2
	wait	S2
	+	X1
	wait	S2
		U1
	+	S2
	+	X2
		U2
B	+	S1
	+	S3
		U1
		U3
D	+	S3
	+	S3
		U3

Завдання 1.3. Для створених історій у 1-му та 2-му пунктів завдань визначте наявність тупика транзакції, створивши граф очікування транзакцій.



Графы для обоих вариантов протоколов идентичные

Завдання 2.1. Встановіть з'єднання з вашою базою даних.

```
tomchuk_viktoriya@vpsj3leQ:~  
[tomchuk_viktoriya@vpsj3leQ ~]$ psql  
psql (9.5.25)  
Type "help" for help.  
tomchuk_viktoriya=>
```

Завдання 2.2. У відповідності із варіантом з таблиці 1 створіть реляційну таблицю в базі даних.

```
tomchuk_viktoriya@vpsj3leQ:~  
tomchuk_viktoriya=> CREATE TABLE employer  
tomchuk_viktoriya-> (e_id integer, name char(20),  
tomchuk_viktoriya-> salary integer);  
CREATE TABLE  
tomchuk_viktoriya=>
```

Завдання 2.3. У відповідності із варіантом з таблиці 1 додайте рядок в реляційну таблицю, створену у попередньому пункті завдання.

```
tomchuk_viktoriya@vpsj3leQ:~  
tomchuk_viktoriya=> INSERT INTO employer VALUES (1, 'Ivanov', 200);  
INSERT 0 1  
tomchuk_viktoriya=>
```

Завдання 2.4. Створіть ще одну операцію внесення рядка в таблицю, який буде відрізнятися значеннями всіх змінних від прикладу з варіанту.

```
tomchuk_viktoriya@vpsj3leQ:~  
tomchuk_viktoriya=> INSERT INTO employer VALUES (2, 'Robin', 150);  
INSERT 0 1  
tomchuk_viktoriya=>
```

Завдання 2.5. Перевіряють вміст таблиці.

```
tomchuk_viktoriya@vpsj3leQ:~  
tomchuk_viktoriya=> SELECT * FROM employer;  
 e_id |      name      | salary  
-----+-----+-----  
    1 | Ivanov         |    200  
    2 | Robin          |    150  
(2 rows)  
tomchuk_viktoriya=>
```

Завдання 3.1. Створіть дві транзакції, кожна з яких повинна включати такі операції:

- операція читання першого рядку таблиці;
- операція зміни однієї із змінних таблиці в першому рядку;
- повторна операція читання першого рядку таблиці;
- операція фіксації всіх змін.

Транзакція 1	Транзакція 2
START TRANSACTION	START TRANSACTION
SELECT * FROM employer WHERE e_id = 1;	SELECT * FROM employer WHERE e_id = 1;
UPDATE employer SET name = 'Admin' WHERE e_id = 1;	UPDATE employer SET salary = 1600 WHERE e_id = 1;
SELECT * FROM employer WHERE e_id = 1;	SELECT * FROM employer WHERE e_id = 1;
COMMIT;	COMMIT;

Завдання 3.2. При створенні транзакцій включіть відповідні операції блокування для протоколу 1-го ступеня блокування.

Транзакція 1	Транзакція 2
START TRANSACTION	START TRANSACTION
SELECT * FROM employer WHERE e_id = 1;	SELECT * FROM employer WHERE e_id = 1;
LOCK TABLE employer IN EXCLUSIVE MODE;	LOCK TABLE employer IN EXCLUSIVE MODE;
UPDATE employer SET name = 'Admin' WHERE e_id = 1;	UPDATE employer SET salary = 1600 WHERE e_id = 1;
SELECT * FROM employer WHERE e_id = 1;	SELECT * FROM employer WHERE e_id = 1;
COMMIT;	COMMIT;

Завдання 3.3. У двох терміналах виконайте операції транзакцій при їх квазіпаралельному режимі роботи за умови, що одна з транзакцій стартує першою.

tomchuk_viktoriya@vpsj3leQ:~

```
START TRANSACTION
tomchuk_viktoriya=> SELECT *
tomchuk_viktoriya-> FROM employer
tomchuk_viktoriya-> WHERE e_id = 1;
e_id |      name      | salary
-----+-----+-----
1 | Ivanov         |    200
(1 row)

tomchuk_viktoriya=> LOCK TABLE employer in EXCLUSIVE MODE;
LOCK TABLE
tomchuk_viktoriya=> UPDATE employer
tomchuk_viktoriya-> SET name = 'Admin'
tomchuk_viktoriya-> WHERE e_id = 1;
UPDATE 1
tomchuk_viktoriya=> SELECT *
tomchuk_viktoriya-> FROM employer
tomchuk_viktoriya-> WHERE e_id = 1;
e_id |      name      | salary
-----+-----+-----
1 | Admin          |    200
(1 row)

tomchuk_viktoriya=>
```

tomchuk_viktoriya@vpsj3leQ:~

```
login as: tomchuk_viktoriya
tomchuk_viktoriya@91.219.60.189's password:
Last login: Mon Apr 26 00:55:06 2021 from 78.26.152.249
[ tomchuk_viktoriya@vpsj3leQ ~ ] $ psql
psql (9.5.25)
Type "help" for help.

tomchuk_viktoriya=> START TRANSACTION;
START TRANSACTION
tomchuk_viktoriya=> SELECT *
tomchuk_viktoriya-> FROM employer
tomchuk_viktoriya-> WHERE e_id = 1;
e_id |      name      | salary
-----+-----+-----
1 | Ivanov         |    200
(1 row)

tomchuk_viktoriya=> LOCK TABLE employer in EXCLUSIVE MODE;
```

```
tomchuk_viktoria@vpsj3leQ:~  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Ivanov | 200  
(1 row)  
  
tomchuk_viktoria-> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoria-> UPDATE employer  
tomchuk_viktoria-> SET name = 'Admin'  
tomchuk_viktoria-> WHERE e_id = 1;  
UPDATE 1  
tomchuk_viktoria-> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Admin | 200  
(1 row)  
  
tomchuk_viktoria=> COMMIT;  
COMMIT  
tomchuk_viktoria=> █  
  
tomchuk_viktoria@vpsj3leQ:~  
START TRANSACTION  
tomchuk_viktoria=> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Ivanov | 200  
(1 row)  
  
tomchuk_viktoria=> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoria=> UPDATE employer  
tomchuk_viktoria-> SET salary = '1600'  
tomchuk_viktoria-> WHERE e_id = 1;  
UPDATE 1  
tomchuk_viktoria=> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Admin | 1600  
(1 row)  
tomchuk_viktoria=> █
```

Завдання 3.4. Повторіть роботу транзакцій, але в першій транзакції замість операції фіксації виконайте операцію відміни.

```
tomchuk_viktoria@vpsj3leQ:~  
tomchuk_viktoria=> START TRANSACTION;  
START TRANSACTION  
tomchuk_viktoria=> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Admin | 1600  
(1 row)  
  
tomchuk_viktoria=> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoria=> UPDATE employer  
tomchuk_viktoria-> SET name = 'Admin'  
tomchuk_viktoria-> WHERE e_id = 1;  
UPDATE 1  
tomchuk_viktoria=> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Admin | 1600  
(1 row)  
  
tomchuk_viktoria=> ROLLBACK;  
ROLLBACK  
tomchuk_viktoria=> █  
  
tomchuk_viktoria@vpsj3leQ:~  
START TRANSACTION  
tomchuk_viktoria=> SELECT *  
tomchuk_viktoria-> FROM employer  
tomchuk_viktoria-> WHERE e_id = 1;  
e_id | name | salary  
-----+-----+-----  
1 | Admin | 1600  
(1 row)  
  
tomchuk_viktoria=> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoria=> █
```

Завдання 3.5. Повторіть пункти 3 та 4 але з використанням протоколу 2-го ступеня блокування.

```
tomchuk_viktoriya@vpsj3leQ:~  
tomchuk_viktoriya-> WHERE e_id = 1;  
e_id |          name          | salary  
-----+-----+-----  
1 | Admin                  | 1600  
(1 row)  
  
tomchuk_viktoriya=> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoriya=> UPDATE employer  
tomchuk_viktoriya=> SET name = 'Admin'  
tomchuk_viktoriya-> WHERE e_id = 1;  
UPDATE 1  
tomchuk_viktoriya=> SELECT *  
tomchuk_viktoriya-> FROM employer  
tomchuk_viktoriya-> WHERE e_id = 1;  
e_id |          name          | salary  
-----+-----+-----  
1 | Admin                  | 1600  
(1 row)  
  
tomchuk_viktoriya=> ROLLBACK;  
ROLLBACK  
tomchuk_viktoriya=> START TRANSACTION  
tomchuk_viktoriya-> ;  
START TRANSACTION  
tomchuk_viktoriya=> LOCK TABLE employer in SHARE MODE;  
  
tomchuk_viktoriya@vpsj3leQ:~  
START TRANSACTION  
tomchuk_viktoriya=> SELECT *  
tomchuk_viktoriya-> FROM employer  
tomchuk_viktoriya-> WHERE e_id = 1;  
e_id |          name          | salary  
-----+-----+-----  
1 | Admin                  | 1600  
(1 row)  
  
tomchuk_viktoriya=> LOCK TABLE employer in EXCLUSIVE MODE;  
LOCK TABLE  
tomchuk_viktoriya=> 
```

tomchuk_viktoriya@vpsj3leQ:~

(1 row)

```
tomchuk_viktoriya=> LOCK TABLE employer in EXCLUSIVE MODE;
LOCK TABLE
```

```
tomchuk_viktoriya=> UPDATE employer
tomchuk_viktoriya-> SET name = 'Admin'
tomchuk_viktoriya-> WHERE e_id = 1;
UPDATE 1
```

```
tomchuk_viktoriya=> SELECT *
tomchuk_viktoriya-> FROM employer
tomchuk_viktoriya-> WHERE e_id = 1;
  e_id |      name      | salary
-----+-----+-----
      1 | Admin          |    1600
(1 row)
```

```
tomchuk_viktoriya=> ROLLBACK;
ROLLBACK
```

```
tomchuk_viktoriya=> START TRANSACTION
tomchuk_viktoriya-> ;
START TRANSACTION
```

```
tomchuk_viktoriya=> LOCK TABLE employer in SHARE MODE;
LOCK TABLE
```

```
tomchuk_viktoriya=> UPDATE employer
tomchuk_viktoriya-> SET name = 'NoAdmin'
tomchuk_viktoriya-> WHERE e_id = 1;
```

tomchuk_viktoriya@vpsj3leQ:~

```
LOCK TABLE
```

```
tomchuk_viktoriya=> ROLLBACK;
ROLLBACK
```

```
tomchuk_viktoriya=> LOCK TABLE employer in SHARE MODE;
ERROR:  LOCK TABLE can only be used in transaction blocks
tomchuk_viktoriya=> START TRANSACTION;
START TRANSACTION
```

```
tomchuk_viktoriya=> LOCK TABLE employer in SHARE MODE;
LOCK TABLE
```

```
tomchuk_viktoriya=> UPDATE employer
tomchuk_viktoriya-> SET salary = '0'
tomchuk_viktoriya-> WHERE e id = 1;
```

tomchuk_viktoriya@vpsj3leQ:~

```
START TRANSACTION
```

```
tomchuk_viktoriya=> LOCK TABLE employer in SHARE MODE;
LOCK TABLE
```

```
tomchuk_viktoriya=> UPDATE employer
tomchuk_viktoriya-> SET name = 'NoAdmin'
tomchuk_viktoriya-> WHERE e_id = 1;
UPDATE 1
tomchuk_viktoriya=>
```

tomchuk_viktoriya@vpsj3leQ:~

```
LINE 1: UPDATE employer
```

```
DETAIL:  Process 9958 waits for RowExclusiveLock on relation 16633 of database 1
6544; blocked by process 9954.
```

```
Process 9954 waits for RowExclusiveLock on relation 16633 of database 16544; blo
cked by process 9958.
```

```
HINT:  See server log for query details.
```

```
tomchuk_viktoriya=>
```

ВИСНОВОК: Вдалося дослідити поведінку процесів-транзакцій в базах даних та засоби керування ними через механізм блокування з використанням сучасних систем керування базами даних. Як можна побачити, друга транзакція була заблокована першою.