

ТРАНЗАКЦІЇ

Лекція 7 курсу «ОБД» Автор Голуб Б.Л.

2024 р.

Визначення транзакції

Транзакція – це сукупність дій, які виконуються по відношенню до БД та розглядаються як єдине ціле. Транзакції – це одиниці активності або, інакше кажучи, послідовність дій, що виконуються за певним логічним порядком. Вони можуть виконуватися як вручну, так і автоматично за допомогою спеціальних програм. В реляційних БД транзакції здійснюються за допомогою, так званих, команд DML – INSERT, UPDATE, DELETE.

Властивості транзакції

- Транзакція - логічна одиниця роботи в базі даних
- Транзакція - одиниця відновлення інформації при аварійній ситуації СУБД
- При фіксації змін у базі даних гарантується збереження або всіх змін, або жодної
- Виконуються всі правила й перевірки, що забезпечують цілісність даних

ACID для транзакцій

- **Неподільність** (*Atomicity*). Транзакція або виконується повністю, або не виконується.
- **Погодженість** (*Consistency*). Транзакція переводить базу даних з одного погодженого стану в інше.
- **Ізольованість** (*Isolation*). Результати транзакції стають доступні для інших транзакцій тільки після її фіксації.
- **Тривалість** (*Durability*). Після фіксації транзакції зміни стають постійними

Правила управління транзакціями

- Кожна транзакція має свій початок і своє завершення.
- Будь-яку транзакцію можна або зберегти, або відмінити.
- Якщо у будь-якому місці виконання транзакції один з операторів не може бути виконаний, вся транзакція є неуспішною і не може бути збережена у БД.

Типи транзакцій

- Явні транзакції. За замовчуванням, кожна команда виконується як окрема транзакція. Користувач може об'єднати кілька команд в одну транзакцію, явно вказавши її початок і кінець.
- Неявні транзакції. Не існує оператора початку транзакції. Транзакція починається з початком сеансу роботи із БД. Завершується транзакція при наступних подіях:
 - ❖ Явно виконаний оператор завершення транзакції - *rollback* або *commit*
 - ❖ Оператор DDL (**Data Definition Language**)
 - ❖ Завершення сеансу
 - ❖ Після закінчення транзакції відразу неявно починається нова транзакція

Початок транзакції

```
BEGIN TRAN [ SACTION ] [ transaction_name  
| @tran_name_variable  
  [ WITH MARK [ 'description' ] ] ]
```

Команди управління транзакціями

- **COMMIT (COMMIT WORK).** Оператор COMMIT завершує транзакцію й робить будь-які виконані в ній зміни постійними (тривалими). У розподілених транзакціях використовуються розширення оператора COMMIT. Ці розширення дозволяють позначити оператор COMMIT.
- **ROLLBACK (ROLLBACK WORK).** Простий оператор повернення завершує транзакцію й скасовує всі виконані в ній і незафіксовані зміни.
- **SAVEPOINT.** Оператор SAVEPOINT дозволяє створити в транзакції "мітку", або крапку збереження. В одній транзакції можна виконувати оператор SAVEPOINT кілька разів, установлюючи кілька крапок збереження.
- **ROLLBACK TO <крапка збереження>.** Цей оператор використовується разом із представленим вище оператором SAVEPOINT. Транзакцію можна повернути до зазначеної крапки збереження, не скасовуючи всі зроблені до неї зміни.
- **SET TRANSACTION.** Цей оператор дозволяє встановлювати атрибути транзакції, такі як рівень ізоляваності й те, як вона буде використовуватися: тільки для читання даних або для читання й запису.

Завершення транзакції

COMMIT [TRAN [SACTION] [*transaction_name* | *@tran_name_variable*]]

Відміна транзакції

ROLLBACK [TRAN [SACTION]

[*transaction_name* | *@tran_name_variable*
| *savepoint_name* | *@savepoint_variable*]

]

Проміжне збереження результатів

SAVE TRAN [SACTION]

{ *savepoint_name* | *@savepoint_variable* }

Одна транзакція

```
BEGIN TRANSACTION  
USE pubs  
GO  
UPDATE titles  
SET advance = advance * 1.25  
WHERE ytd_sales > 8000  
GO  
COMMIT  
GO
```

Вкладені транзакції - 1

```
CREATE TABLE TestTran (Cola INT PRIMARY KEY,  
Colb CHAR(3))  
GO
```

Вкладені транзакції -2

```
BEGIN TRANSACTION OuterTran
```

```
-- @@TRANCOUNT set to 1.
```

```
GO
```

```
INSERT INTO TestTran VALUES (1,  
'aaa')
```

```
GO
```

Вкладені транзакції -3

```
BEGIN TRANSACTION Inner1
-- @@TRANCOUNT set to 2.
GO
INSERT INTO TestTran VALUES
(2, 'bbb')
GO
```

Вкладені транзакції -4

```
BEGIN TRANSACTION Inner2
```

```
-- @@TRANCOUNT set to 3.
```

```
GO
```

```
INSERT INTO TestTran VALUES (3,  
'ccc')
```

```
GO
```

Вкладені транзакції -5

COMMIT TRANSACTION Inner2

- Decrements @@TRANCOUNT to 2.
- Nothing committed.

GO

COMMIT TRANSACTION Inner1

- Decrements @@TRANCOUNT to 1.
- Nothing committed.

GO

COMMIT TRANSACTION OuterTran

- Decrements @@TRANCOUNT to 0.
- Commits outer transaction OuterTran.

GO