**ЛАБОРАТОРНА РОБОТА №19**

ЗАСТОСУВАННЯ ТРИГЕРІВ В T-SQL

*Мета:* основна мета цієї лабораторної роботи полягає у наданні студентам глибокого розуміння принципів роботи, створення та використання тригерів у Transact-SQL, які використовуються в середовищі Microsoft SQL Server. Тригери є особливим типом збережених процедур, які автоматично запускаються в базі даних у відповідь на певні події, такі як вставка, оновлення чи видалення записів.

Детальні цілі лабораторної роботи включають:

1. Освоєння концепцій тригерів: студенти повинні вивчити основні концепти тригерів, включно з їх класифікацією на DML-тригери (Data Manipulation Language) і DDL-тригери (Data Definition Language). Завдання передбачає вивчення механізмів запуску тригерів та відповідних подій, які їх ініціюють.

2. Розробка та тестування тригерів: практичні навички з написання тригерів, що включає сценарії їх застосування для аудиту змін у базі даних, забезпечення цілісності даних та автоматичної реакції на зміни стану даних. Студенти навчаться розробляти тригери, які ефективно обробляють виключення та помилки.

3. Аналіз впливу тригерів на продуктивність системи: дослідження потенційного впливу тригерів на продуктивність бази даних. Завдання включає аналіз сценаріїв, в яких використання тригерів може знижувати чи покращувати продуктивність, з особливим фокусом на оптимізацію та потенційні недоліки.

4. Розробка стратегій управління тригерами: оволодіння методами управління життєвим циклом тригерів, включаючи їх активацію, деактивацію, моніторинг та налагодження. Студенти дізнаються, як адекватно планувати впровадження тригерів у виробничі системи, забезпечуючи мінімізацію непередбачуваних наслідків для бізнес-операцій.

**ТЕОРЕТИЧНІ ОСНОВИ**

Тригери в T-SQL є спеціальними процедурами, що автоматично виконуються у відповідь на певні події в базі даних, такі як вставка, оновлення або видалення даних. Вони діють як потужні інструменти для забезпечення цілісності даних, реалізації бізнес-логіки на рівні бази даних та аудиту змін у базі даних.

**Основні види тригерів в T-SQL:**

1. DML тригери (Data Manipulation Language):

- AFTER тригери (відомі також як FOR тригери): Виконуються після того, як DML-операція (INSERT, UPDATE, DELETE) вже відбулась, що дозволяє їм реагувати на зміни, вже здійснені в базі даних.

- INSTEAD OF тригери: Запускаються замість оригінальної DML-операції, дозволяючи переписати стандартну поведінку операції.

2. DDL тригери (Data Definition Language):

- Виконуються у відповідь на зміни схеми бази даних, такі як створення або зміна об'єктів бази даних (таблиць, процедур тощо).

**Приклади застосування тригерів:**

- Аудит змін: Автоматичне ведення журналу змін в таблиці. Наприклад, AFTER UPDATE тригер може записувати попередні значення та нові значення змінених рядків у спеціальну таблицю аудиту.

CREATE TRIGGER AuditTrigger

ON Employees

AFTER UPDATE

AS

BEGIN

INSERT INTO AuditTable (UserID, OldValue, NewValue, ChangeDate)

SELECT USER\_ID(), deleted.\*, inserted.\*, GETDATE()

FROM inserted

JOIN deleted ON inserted.EmployeeID = deleted.EmployeeID;

END;

- Забезпечення бізнес-правил: INSTEAD OF тригер може перевіряти, чи задовольняють введені дані певним умовам, перш ніж дозволити оновлення таблиці.

CREATE TRIGGER ValidateEmployee

ON Employees

INSTEAD OF UPDATE

AS

BEGIN

IF EXISTS (SELECT \* FROM inserted WHERE Salary < 30000)

BEGIN

RAISERROR ('Salary cannot be less than 30000.', 16, 1);

ROLLBACK TRANSACTION;

END

ELSE

BEGIN

UPDATE Employees

SET Name = inserted.Name, Salary = inserted.Salary

FROM inserted

WHERE Employees.EmployeeID = inserted.EmployeeID;

END

END;

***Особливості застосування тригерів:***

- Продуктивність: Тригери можуть впливати на продуктивність системи, особливо якщо вони виконують складні операції або з'являються на таблицях з великою кількістю транзакцій.

- Складність управління: Неправильне використання тригерів може призвести до складних залежностей і неочікуваних результатів, особливо в складних системах з багатьма тригерами.

- Тестування та налагодження: Тригери потребують ретельного тестування та налагодження, оскільки вони можуть бути активовані автоматично, що ускладнює процес ідентифікації та вирішення помилок.

**ЗАВДАННЯ**

***Завдання 1***

Реалізуйте тригер для перевірки чисельності працівників на посаді.

Необхідно, щоб тригер блокував транзакцію (додавання нового працівника) у випадку, якщо найм працівника перевищить ліміт (не більше 8 працівників на кожній окремій посаді).

- Використовується таблиця dbo.salesreps

- Задійніть директиву INSTEAD OF INSERT

***Завдання 2***

Тригер для автоматичного оновлення дати модифікації

Мета: автоматичне оновлення поля `LastModified` у таблиці `Products` при будь-яких змінах даних про продукти.

Умови завдання:

- до таблиці PRODUCTS додайте новий стовбець `LastModified` (тип даних – дата, може мати null значення)

- Створіть тригер, який реагує на зміни в таблиці `Products`, зокрема на операції UPDATE.

- Тригер має автоматично встановлювати поточну дату і час в поле `LastModified` для рядка, що зазнав змін.

- Використовуйте системну функцію `GETDATE()` для отримання поточної дати та часу.

***Завдання 3***

Тригер для перевірки унікальності електронної пошти

Запобігання дублюванню адрес електронної пошти в таблиці `Customers`.

Умови завдання:

- до таблиці CUSTOMERS додайте новий стовбець `Email` (може мати null значення)

- Створіть тригер, який перевіряє унікальність електронної пошти при додаванні нових або оновленні існуючих записів у таблиці `Customers`.

- Тригер повинен блокувати вставку або оновлення запису, якщо введена адреса електронної пошти вже існує в базі даних.

- Використовуйте директиву `INSTEAD OF INSERT, UPDATE` для перехоплення спроб вставки або оновлення.

**ДОДАТКОВІ МАТЕРІАЛИ:**

<https://learn.microsoft.com/ru-ru/sql/t-sql/statements/create-trigger-transact-sql?view=sql-server-ver16>

**КОНТРОЛЬНІ ЗАПИТАННЯ**

1. Що таке тригер у контексті баз даних SQL Server? Опишіть визначення тригера, зазначте його основні функції та перелічіть типи подій, які можуть активувати тригер.

2. Яка різниця між AFTER тригерами та INSTEAD OF тригерами? Обговоріть, як кожен тип тригера впливає на обробку транзакцій і в яких ситуаціях кожен тип краще використовувати.

3. Які потенційні небезпеки пов'язані з використанням тригерів? Поясніть, як неправильне або надмірне використання тригерів може призвести до зниження продуктивності або ускладнення відладки системи баз даних.

4. Наведіть приклад тригера, який використовується для забезпечення цілісності даних. Задайте сценарій, в якому тригер перевіряє дотримання бізнес-правил або обмежень інтегритету перед зміною даних у таблиці.

5. Як можна використовувати тригери для автоматичного ведення аудиту змін даних? Опишіть процес створення тригера, який автоматично записує інформацію про зміни, здійснені в таблицях бази даних, включаючи тип зміни, користувача, який здійснив зміну, та час зміни.

6. Як тригери можуть бути використані для оптимізації взаємодій з клієнтами в бізнес-додатках? Розгляньте сценарій, в якому тригери використовуються для автоматичного відправлення сповіщень або виконання інших дій у відповідь на певні транзакції або події в базі даних.

7. Обговоріть стратегії оптимізації продуктивності при використанні тригерів. Які методи можна застосувати для зменшення негативного впливу тригерів на продуктивність бази даних? Наведіть приклади налаштувань, які можуть допомогти у цьому процесі.