



Національний технічний університет України  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
Фізико-Технічний Інститут

Лабораторна робота №8  
Варіант -16

Виконав:  
студент групи ФІ-73  
Чіхладзе Вахтанг

Викладач:  
Коломицев Михайло Володимирович

Київ 2020

Варіант № 16

Предметна область: Фільмотека .

Основні предметно-значимі сутності: фільми, актори.

Основні предметно-значимі атрибути сутності:

- фільми – назва фільму, жанр, тривалість, рік виходу, стрічка (кол. чи ч.б.).

- актори – прізвище, в яких фільмах грав, дата народження; стать

Основні вимоги до функцій системи:

- вибрати всіх акторів, що грали у фільмі вказаного жанру;
- підрахувати кількість фільмів, зіграних кожним актором;
- визначити актора, що зіграв найбільше ролей за останні 10 років.

Тригери:

1. На видалення запису з таблиці «Фільми». Якщо з цим фільмом пов'язані записи в таблиці «АкториФільми», видалити і ці записи.

2. На додавання / оновлення записів в таблиці «АкториФільми». Створити представлення «ІндексАктивності» з полями «код\_актора», «ім'я», «Прізвище», «кількість\_фільмів», «рік\_виходу», де поле «кількість\_фільмів» розраховується на кожний рік, коли знімався даний актор. При додаванні запису в таблицю «АкториФільми» і при оновленні запису, якщо поле «рік\_виходу» змінювалось, оновлювати вид «ІндексАктивності».

Збережена процедура:

Процедура повинна повертати кількість фільмів конкретного жанру.

Ціль заняття: Знайомство з об'єктами СХЕМИ БАЗИ ДАНИХ. Створення і використання тригерів.

Результат заняття: Текст тригера у відповідності до наведеного завдання. Текст тригера контролю цілісності даних для варіанта завдання.

Завдання до лабораторної роботи:

Створити тригери, які вказані у вашому завданні.

Хід роботи

USE FilmLibraryDatabase

GO

CREATE TRIGGER deletefilm

ON films FOR DELETE

AS

DECLARE @count INT

SELECT @count=COUNT(\*) FROM deleted D

IF @count>0

BEGIN

ALTER TABLE movieMaking NOCHECK CONSTRAINT FK\_\_movieMakin\_\_film\_\_6E01572D

DELETE FROM movieMaking

FROM deleted D JOIN movieMaking M ON D.film\_id=M.film

WHERE D.film\_id=M.film

ALTER TABLE movieMaking WITH CHECK CHECK CONSTRAINT

FK\_\_movieMakin\_\_film\_\_6E01572D

END

```
GO
CREATE VIEW indexActivity
AS
SELECT A.actor_id,A.name,F.film_year,count(A.actor_id)[кількість фільмів за рік]
FROM movieMaking M LEFT JOIN films F ON M.film=F.film_id
LEFT JOIN actors A ON M.actor=A.actor_id
GROUP BY A.actor_id,A.name,F.film_year
```

Створювати тригер на додавання або оновлення таблиці «АкториФільми» не є доцільним, оскільки представлення indexActivity є ссилкою на таблиці з нашої бази, яке виконується динамічно.

Висновок: Отже, я навчився працювати з тригерами та реалізував тригер на видалення запису в таблиці фільми.