



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

Лабораторна робота №3
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: вивчення синтаксису мови маніпулювання даними (Data Manipulation

Language - DML).

Результат заняття: заповнення таблиць бази даних у відповідності до варіанта завдання.

Завдання для лабораторної роботи

1. Використовуючи команду INSERT заповнити таблиці бази даних значеннями. Кількість рядків в таблицях – не менше 10.

Хід роботи

З лабораторної роботи №1 маємо:

USE FilmLibraryDatabase

GO

DROP TABLE genre,footage,sex,films,actors,movieMaking

```
CREATE TABLE genre(  
  genre_id INT PRIMARY KEY IDENTITY NOT NULL,  
  genre_name VARCHAR(15) NOT NULL  
);
```

```
CREATE TABLE footage(  
  footage_id INT PRIMARY KEY IDENTITY NOT NULL,
```

```
footage_type VARCHAR(15) NOT NULL  
);
```

```
CREATE TABLE sex(  
sex_id INT PRIMARY KEY IDENTITY NOT NULL,  
sex VARCHAR(10) NOT NULL  
);
```

```
CREATE TABLE films(  
film_id INT PRIMARY KEY IDENTITY NOT NULL,  
film_name VARCHAR(50) NOT NULL,  
film_year DATE NOT NULL,  
duration TIME NOT NULL,  
genre INT FOREIGN KEY REFERENCES genre(genre_id),  
footage INT FOREIGN KEY REFERENCES footage(footage_id)  
);
```

```
CREATE TABLE actors(  
actor_id INT PRIMARY KEY IDENTITY NOT NULL,  
name VARCHAR(30) NOT NULL,  
birth DATE NOT NULL,  
sex INT FOREIGN KEY REFERENCES sex(sex_id)  
);
```

```
CREATE TABLE movieMaking(  
film INT FOREIGN KEY REFERENCES films(film_id),  
actor INT FOREIGN KEY REFERENCES actors(actor_id)  
);
```

Тепер створимо данні таким скриптом:

```
USE FilmLibraryDatabase  
GO
```

```
DELETE FROM footage  
DELETE FROM sex  
DELETE FROM genre  
DELETE FROM films  
DELETE FROM actors  
DELETE FROM movieMaking
```

```
SET IDENTITY_INSERT footage OFF  
SET IDENTITY_INSERT sex OFF  
SET IDENTITY_INSERT genre OFF  
SET IDENTITY_INSERT films OFF
```

```
SET IDENTITY_INSERT actors OFF
```

```
INSERT INTO footage(footage_type) VALUES  
( 'black-white'),  
( 'colored')
```

```
INSERT INTO sex(sex) VALUES  
( 'male'),  
( 'female')
```

```
INSERT INTO genre(genre_name) VALUES  
( 'action'),  
( 'drama'),  
( 'comedy'),  
( 'triller'),  
( 'horror')
```

```
INSERT INTO films(film_name,film_year,duration,genre,footage) VALUES  
( 'John Wick','2014-10-24','01:41:00',1,2),  
( 'John Wick 2','2017-01-30','01:41:00',1,2),  
( 'John Wick 3','2019-05-09','02:01:00',1,2)
```

```
INSERT INTO actors(name,birth,sex) VALUES  
( 'Keanu Reeves','1964-09-02',1),  
( 'Michael Nyqvist','1960-10-8',1),  
( 'Ruby Rose','1986-03-20',2)
```

```
INSERT INTO movieMaking(film,actor) VALUES  
(1,1),  
(2,1),  
(3,1),  
(1,2),  
(2,3)
```

Висновок:Отже при виконанні данної лабораторної роботи я отримав навички із заповненням бази даних даними за допомогою команди INSERT INTO.А також дізнався про операції DML.