



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

Лабораторна робота №6  
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 представлень.

Результат заняття: Створення представлень у відповідності до наведених завдань. Створення представлень для обраного варіанта лабораторних робіт.

Завдання для лабораторної роботи

Створити представлення для перегляду бази даних з ціллю перегляду інформації, сформульованої в розділі «основні вимоги щодо функцій системи» завдання.

Хід роботи

```
CREATE VIEW ActorsView
```

```
AS
```

```
SELECT A.name[Им`я актера],F.film_name[Назва фільму],G.genre_name[Жанр]
```

```
FROM movieMaking M
```

```
INNER JOIN actors A ON M.actor=A.actor_id
```

```
INNER JOIN films F ON M.film=F.film_id
```

```
INNER JOIN genre G ON F.genre=G.genre_id
```

```
WHERE F.genre=1
```

```
CREATE VIEW howManyFilmsActorPlayed
```

```
AS
```

```
SELECT count(M.actor)[кількість фільмів зіграних актором],A.name[ім`я актора]
```

```
FROM movieMaking M
```

```
INNER JOIN actors A ON M.actor=A.actor_id
```

```
GROUP BY M.actor,A.name
```

```
CREATE VIEW topActor
```

AS

SELECT TOP(1) A.name[ім`я актора],count(M.actor)[зіграв найбільше ролей за останні 10 років]

FROM movieMaking M

JOIN actors A ON M.actor=A.actor\_id

JOIN films F ON M.film=F.film\_id

WHERE F.film\_year BETWEEN CONVERT(DATE,DATEADD(year,-10,GETDATE())) and

CONVERT(DATE,GETDATE())

GROUP BY M.actor,A.name

ORDER BY count(M.actor) DESC

Висновок:Отже,я навчився працювати створювати представлення та робити запити по представленням.