## КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ КАФЕДРА ПРОГРАМНИХ СИСТЕМ І ТЕХНОЛОГІЙ**

**Лабораторна робота №5**

з дисципліни «Архітектура та проектування баз даних»

## Варіант №4

**«Сховище, в якому зберігається інформація про художні фільми»**

Виконав студент групи: ІПЗ-23,

Гоша Давід

Перевірила: Духновська Ксенія Костянтинівна

## Київ – 2022

**Тема:**

Оновлення даних. Видалення даних

# Завдання №1

За своїм варіантом та за прикладом створити словник своєї предметної області.

## Словник предметної області:

*Відношення Фільми:*

|  |  |  |
| --- | --- | --- |
| Назва | Тип, довжина | Ключ, обмеження |
| Id |  | PK |
| Назва фільму | Nvarchar(100) | FK |
| Країна виробник | Nvarchar(50) | FK |
| Жанр | Nvarchar(MAX) | може мати декілька жанрів |
| Тривалість | Nchar(12) | hour + min, разом 12 |
| Рік створення | Nchar(4) | тільки рік, без дня та місяця |

*Відношення Назва фільму:*

|  |  |  |
| --- | --- | --- |
| Назва | Тип даних | Ключ |
| Назва фільму | Nvarchar(50) | PK |
| Коротенький опис | String | До 100 слів |
| Назва англійською | Nvarchar(50) |  |

*Відношення Країни:*

|  |  |  |
| --- | --- | --- |
| Назва | Тип даних | ключ |
| Назва країни | Nvarchar(50) | FK країни |
| Мова країни-виробника | Nvarchar(50) | FK мови |
| Мови перекладу | Nvarchar(50) | Може бути декілька мов |
| Рекомендовані фільми цих країн | \* | Посилання на інші фільми цьої країни |

*Відношення Купівлі:*

|  |  |  |
| --- | --- | --- |
| Назва | Тип, довжина | Ключ, обмеження |
| Id товару | Int | РК |
| Ціна | Money | FK, обмеження по uint\_t |
| Власник | Nvarchar(50) | Тільки авторизовані користувачі |

# Завдання №2

Написати запити, за прикладами наведеними в теоретичній частині.

1. Функція replaceOne():

*db.Typ.replaceOne({FilmName: "Cruella"}, {name: "One at Home", Year: "1999"})*

В даному випадку знаходимо документ, в якому FilmName: "Cruella", та замінюємо його документом {name: "One at Home", Year: "1999"}.

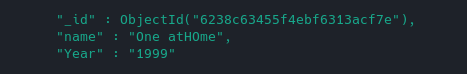
Повідомлення після написання запиту :



До застосування функції replaceOne():



Результат після застосування функції replaceOne():



1. Функція updateOne():

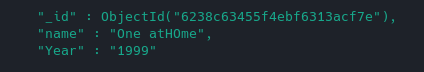
*db.Typ.updateOne({"name" : "One atHOme"}, {$set: {name: "updateOne", Year : "2000"}})*

Тут ми шукаємо документ з *name" : "One atHOme"* та встановлюємо для його властивості *name* значення *updateOne* та для *Year* значення *2000*.

Повідомлення після написання запиту:



До застосування функції updateOne():



Результат після застосування функції updateOne():



1. Оператор $set:

*db.Typ.updateOne({name : "updateOne", year: "2000"}, {$set: {seed: 100/50}})*

Для оновлення окремих полів у цих функціях використовується оператор $set. Якщо документ не містить оновлюване поле, воно створюється. В нашому випадку утворилося поле seed.

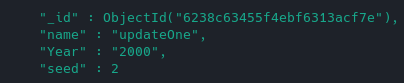
Повідомлення після написання запиту:



До застосування оператора $set:



Результат після застосування оператору $set:



1. Функція updateMany():

*db.Typ.updateMany({year : "2000}, {$set: {country : "Italy"}})*

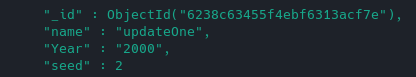
Тут оновлюємо всі документи, котрі відповідають певному критерію (year : "2021"), та встановлюємо, наприклад, іншу країну виробника {$set:

{country : "Italy"}}.

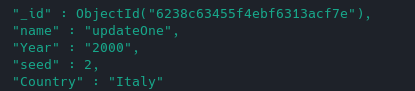
Повідомлення після написання запиту:



До застосування функції updateMany():



Результат після застосування функції updateMany():



5) Оператор $unset:

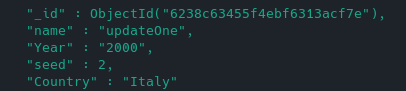
db.Typ.update({name : "updateOne"}, {$unset: {seed : 2}})

Цей оператор призначений для видалення окремого ключа. В нашому випадку - поле, яке ми створили раніше (seed) - видаляється.

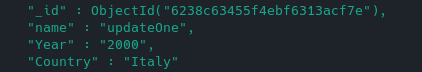
Повідомлення після написання запиту:



До застосування оператора $unset:



Результат після застосування оператора $unset:



1. Оператор $push:

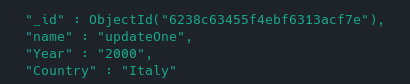
db.Typ.updateOne({name : "updateOne"}, {$push: {Genre: "Thriller"}})

Оператор $push дозволяє додати ще одне значення до існуючого. В нашому випадку до фільму Don`t look up додався жанр Thriller.

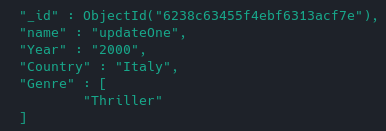
Повідомлення після написання запиту:



До застосування оператора $push:



Результат після застосування оператора $push:



1. Оператори $each, $position, $slice:

*db.Typ.updateOne({name : "updateOne"}, {$push: {Genre: {$each: ["Action", "Comedy"], $position:1, $slice:4}}})*

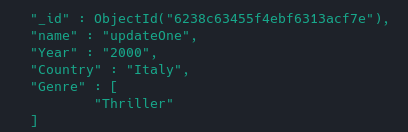
Використовуючи оператор $each, можна додати відразу кілька значень. Оператор $position задає позицію у масиві для вставки елементів, а оператор

$slice вказує, скільки елементів залишити у масиві після вставки. В нашому випадку ми додали одразу додаткові два жанри ["Action", "Comedy"], розташували їх посередині, вивели 4 цих жанри.

Повідомлення після написання запиту:



До застосування оператораторів $each, $position, $slice:



Результат після застосування операторів $each, $position, $slice:



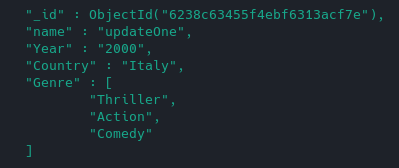
1. Оператор $addToSet:

db.Typ.update({name : "updateOne"}, {$addToSet: {Genre: "Adventures"}})

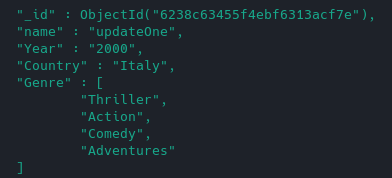
Оператор $addToSet додає об'єкти в масив. Відмінність від $push полягає в тому, що $addToSet додає дані, якщо їх ще немає в масиві. В нашому випадку ми спочатку хотіли додати жанр, який вже є в масиві до цього фільму (Adventures), а його повторно не додало. Повідомлення після написання запиту:



До застосування операторатора $addToSet:



Результат після застосування оператора $addToSet:



1. Оператор $pop:

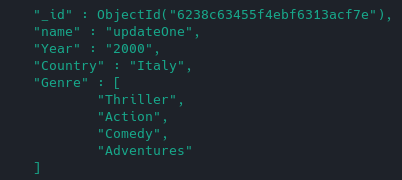
db.Typ.updateOne({name : "updateOne"}, {$pop: {Genre: -1}})

Оператор $pop дозволяє видаляти елемент з масиву. Вказавши на ключ genre: 1, ми видалили останній елемент. Щоб видалити перший елемент, треба написати -1.

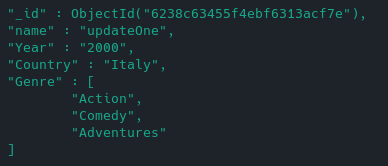
Повідомлення після написання запиту:



До застосування операторатора $pop:



Результат після застосування оператора $pop:



1. Функція deleteOne():

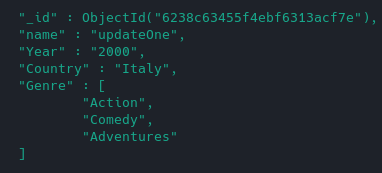
db.Typ.deleteOne({name : "updateOne"})

Функція deleteOne() передбачена для видалення документа, в якому name = "updateOne". Щоб видалити кілька документів, можно використати deleteMany().

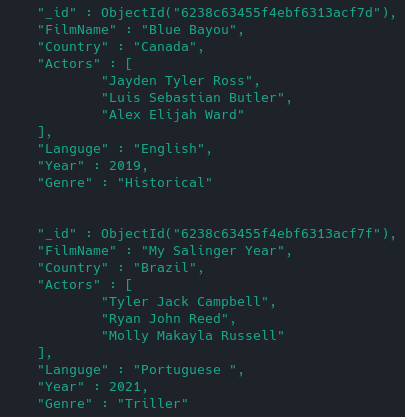
Повідомлення після написання запиту:



До застосування фунції deleteOne():



Після застосування фунції deleteOne():



1. Видалення за допомогою регулярних виразів:

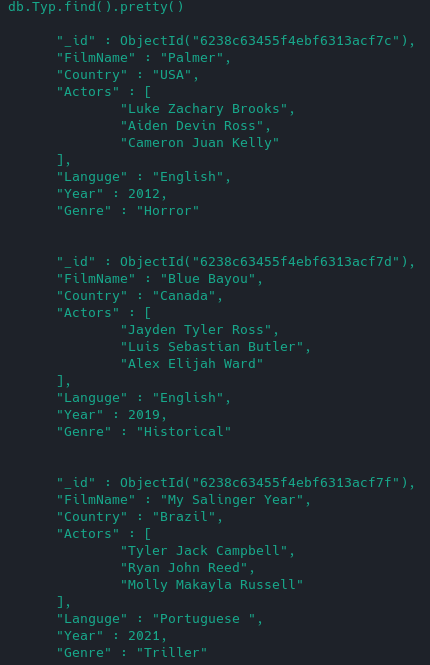
*db.films.deleteOne({name : /^A\w+/i})*

Можливо видалити документ, в якому назва фільму починається з будь-якої літери (в нашому випадку = А).

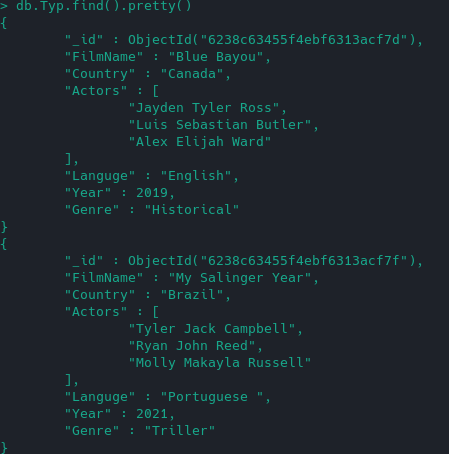
Повідомлення після написання запиту:



До видалення, за допомогою регулярних виразів:



Після видалення, за допомогою регулярних виразів:



# Висновок :

Виконавши лабораторну роботу № 5 на тему: «Оновлення даних. Видалення даних» було створено словник своєї предметної області та написано запити, за прикладами наведеними в теоретичній частині.