

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ  
ТАРАСА ШЕВЧЕНКА  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ПРОГРАМНИХ СИСТЕМ І ТЕХНОЛОГІЙ**

**Лабораторна робота №9**

**“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**

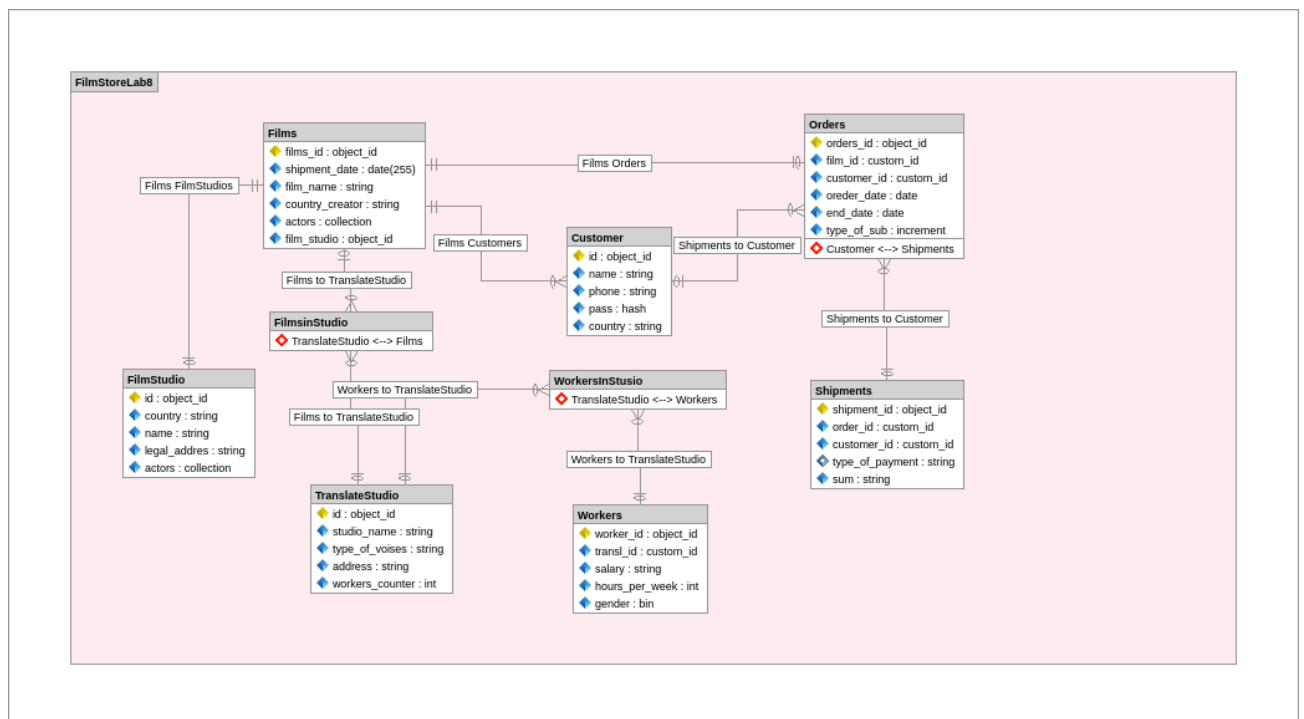
## Тема:

## С# та MongoDB. Case-технології для проектування БД ЗАВДАННЯ

1. За допомогою довільної CASE-технології спроектувати БД для вашого варіанту.
2. Виконати всі приклади із л/р, але застосовуючи їх для вашої предметної області.

## Виконання:

- Спроектувати БД за CASE-технологією



UML Діаграма спроектованої БД

- Виконати всі приклади із л/р

Я вирішив з'єднати всі завдання в 1 проект і ось що у мене вийшло.

Спочатку ми друкуємо усі бд з серверу. Потім ми друкуємо усі конекції цих бд. У третьому пункті ми створюємо модель даних за допомогою BsonDocument, потім за допомогою класу та конвертуємо його в BsonDocument.

Отримання всіх бд із сервера

```
Тип
admin
config
local
test
```

Отримаємо всі колекції всіх баз даних, які є на сервері

У базі даних Тип є наступні колекції:

Тип

У базі даних admin є наступні колекції:

system.version

У базі даних config є наступні колекції:

system.sessions

У базі даних local є наступні колекції:

startup\_log

У базі даних test є наступні колекції:

Тип

Створимо модель даних

```
{ "_id" : ObjectId("000000000000000000000000"), "country" : "Ukraine", "name" : "GasMyas", "legal_address" : "st Avenue, 1", "actors" : { "Name" : "Ryan Mertviy", "gender" : "male" } }
```

Також можна виконати зворотну операцію перетворення об'єкта в BsonDocument

```
{ "_id" : ObjectId("000000000000000000000000"), "country" : "Ukraine", "name" : "GasMyas", "legal_address" : "st Avenue, 1", "actors" : { "Name" : "Ryan Mertviy", "gender" : "male" } }
```

. При створенні документа ми можемо скористатися як стандартним класом C#, так і класом BsonDocument

```
{ "_id" : ObjectId("000000000000000000000000"), "country" : "Ukraine", "name" : "GasMyas", "legal_address" : "st Avenue, 1", "actors" : { "Name" : "Ryan Mertviy", "gender" : "male" } }
```

## Код програми:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using MongoDB.Bson;
using MongoDB.Driver;
using System.Configuration;
using Microsoft.Extensions.Configuration;
using MongoDB.Bson.Serialization;

namespace BD9
{
    class FilmStudio
    {
        public ObjectId Id { get; set; }
        public string country { get; set; }
        public string name { get; set; }
        public string legal_address { get; set; }
        public Actors actors { get; set; }
    }
    class Actors
```

```

{
    public string Name { get; set; }
    public string gender { get; set; }
}
class Program
{
    static void Main(string[] args)
    {
        string connectionString = "mongodb://localhost:27017";

        MongoClient client = new MongoClient(connectionString);
        Console.WriteLine("");
        Console.WriteLine("Отримання всіх бд із сервера");
        Console.WriteLine("");
        GetDatabaseNames(client).GetAwaiter();
        Console.ReadLine();

        Console.WriteLine("");
        Console.WriteLine("Отримаємо всі колекції всіх баз даних, які є на сервері");
        Console.WriteLine("");
        GetCollectionsNames(client).GetAwaiter();
        Console.ReadLine();

        Console.WriteLine("");
        Console.WriteLine("Створимо модель даних");
        Console.WriteLine("");
        SimpleDocument();

        Console.WriteLine("");
        Console.WriteLine(". При створенні документа ми можемо скористатися як стандартним
класом C#, так і класом BsonDocument");
        Console.WriteLine("");
        BSONDocument();

        Console.ReadLine();
    }
    private static async Task GetDatabaseNames(MongoClient client)
    {
        using (var cursor = await client.ListDatabasesAsync())
        {
            var databaseDocuments = await cursor.ToListAsync();
            foreach (var databaseDocument in databaseDocuments)
            {
                Console.WriteLine(databaseDocument["name"]);
            }
        }
    }
    private static async Task GetCollectionsNames(MongoClient client)
    {
        using (var cursor = await client.ListDatabasesAsync())
        {
            var dbs = await cursor.ToListAsync();
            foreach (var db in dbs)
            {
                Console.WriteLine("У базі даних {0} є наступні колекції:", db["name"]);
                IMongoDatabase database = client.GetDatabase(db["name"].ToString());
                using (var collCursor = await database.ListCollectionsAsync())
                {
                    var colls = await collCursor.ToListAsync();
                    foreach (var col in colls)

```

```

        {
            Console.WriteLine(col["name"]);
        }
    }
    Console.WriteLine();
}
}
}
}
private static async Task SimpleDocument()
{
    FilmStudio filmStudio = new FilmStudio { country = "Ukraine", name = "GasMyas",
legal_address = "st Avenue, 1" };
    filmStudio.actors = new Actors { Name = "Ryan Mertviy", gender = "male" };
    Console.WriteLine(filmStudio.ToJson());
    Console.WriteLine("");
    Console.WriteLine("Також можна виконати зворотну операцію перетворення об'єкта в
BsonDocument");
    Console.WriteLine("");
    ObjectToBson(filmStudio);
}
private static async Task BSONDocument()
{
    BsonDocument doc = new BsonDocument
    {
        {"country", "Ukraine"},
        {"name", "GasMyas"},
        {"legal_address", "st Avenue, 1"},
        { "actors",
            new BsonDocument
            {
                {"Name", "Ryan Mertviy" },
                { "gender", "male"}
            }
        }
    };
    FilmStudio p = BsonSerializer.Deserialize<FilmStudio>(doc);
    Console.WriteLine(p.ToJson());
}
private static async Task ObjectToBson(FilmStudio filmStudio)
{
    BsonDocument doc = filmStudio.ToBsonDocument();
    Console.WriteLine(doc);
}
}
}
}
}

```

## Висновок :

У лабораторній роботі номер 9 ми за допомогою довільної CASE-технології спроектували БД для нашого варіанту. Виконали всі приклади із л/р, але застосовуючи їх для своєї предметної області.