

# Лабораторна робота №4.3

## Тема

Змінення вмісту БД.

## Мета

Розробити додаток для завантаження в БД заданої картинки, отримання списку картинок, вибору та видалення заданої.

## Огляд програми

Програма розроблена на мові програмування C# та використовує технологію бази даних SQLite для зберігання та обробки зображень. Консольний інтерфейс надає зручну взаємодію з користувачем.

## Функціонал програми

Програма має такий функціонал:

1. **Завантаження зображення:** Користувач може завантажити зображення в базу даних, вказавши повний шлях до файлу та назву зображення.
2. **Отримання списку зображень:** Користувач може переглянути список зображень, які знаходяться в базі даних, разом з їхніми унікальними ідентифікаторами.
3. **Видалення зображення:** Користувач може видалити зображення з бази даних, вказавши його унікальний ідентифікатор.

## Процес розробки

Для розробки програми було використано мову програмування C# та бібліотеку `Microsoft.Data.Sqlite` для роботи з базою даних SQLite. Програма створює з'єднання з базою даних, перевіряє наявність таблиці "Images" та створює її, якщо вона відсутня. Зображення зберігаються у форматі BLOB (Binary Large Object) прямо в базі даних.

Програма має модульну структуру, що дозволяє виконувати різні операції з базою даних у залежності від вибору користувача.

## Огляд програми

```
using System;  
using System.IO;  
using Microsoft.Data.Sqlite;
```

```

namespace ImageDatabaseApp
{
    class Program
    {
        static void Main(string[] args)
        {
            string dbConnectionString = "Data Source=image_database.db";

            try
            {
                using (SqlConnection connection = new
SqlConnection(dbConnectionString))
                {
                    connection.Open();
                    Console.WriteLine("З'єднання з базою даних SQLite успішно
встановлено.");

                    // Створення таблиці, якщо вона ще не існує
                    CreateImagesTable(connection);

                    while (true)
                    {
                        Console.WriteLine("Оберіть дію:");
                        Console.WriteLine("1. Завантажити зображення");
                        Console.WriteLine("2. Отримати список зображень");
                        Console.WriteLine("3. Видалити зображення");
                        Console.WriteLine("0. Вийти");

                        int choice = Convert.ToInt32(Console.ReadLine());

                        switch (choice)
                        {
                            case 1:
                                UploadImage(connection);
                                break;
                            case 2:
                                GetImageList(connection);
                                break;
                            case 3:
                                DeleteImage(connection);
                                break;
                            case 0:
                                return;
                            default:
                                Console.WriteLine("Невірний вибір. Будь ласка,
спробуйте ще раз.");
                                break;
                        }
                    }
                }
            }
        }
    }
}

```

```

    }
    catch (Exception ex)
    {
        Console.WriteLine("Помилка: " + ex.Message);
    }
}

static void CreateImagesTable(SqliteConnection connection)
{
    string createImagesTable = @"
        CREATE TABLE IF NOT EXISTS Images (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            name TEXT NOT NULL,
            data BLOB NOT NULL
        )";

    ExecuteNonQuery(connection, createImagesTable);
    Console.WriteLine("Таблицю 'Images' успішно створено або вона вже
існує.");
}

static void ExecuteNonQuery(SqliteConnection connection, string query)
{
    using (SqliteCommand command = new SqliteCommand(query, connection))
    {
        command.ExecuteNonQuery();
    }
}

static void UploadImage(SqliteConnection connection)
{
    Console.WriteLine("Введіть повний шлях до зображення:");
    string imagePath = Console.ReadLine();

    byte[] imageBytes = File.ReadAllBytes(imagePath);

    Console.WriteLine("Введіть назву зображення:");
    string imageName = Console.ReadLine();

    string query = "INSERT INTO Images (name, data) VALUES (@name, @data)";

    using (SqliteCommand command = new SqliteCommand(query, connection))
    {
        command.Parameters.AddWithValue("@name", imageName);
        command.Parameters.AddWithValue("@data", imageBytes);

        int rowsAffected = command.ExecuteNonQuery();
        Console.WriteLine($"{rowsAffected} рядків додано.");
    }
}

```

```

static void GetImageList(SqliteConnection connection)
{
    string query = "SELECT id, name FROM Images";

    using (SqliteCommand command = new SqliteCommand(query, connection))
    {
        using (SqliteDataReader reader = command.ExecuteReader())
        {
            Console.WriteLine("Список зображень:");

            while (reader.Read())
            {
                Console.WriteLine($"ID: {reader.GetInt32(0)}, Назва:
{reader.GetString(1)}");
            }
        }
    }

    static void DeleteImage(SqliteConnection connection)
    {
        Console.WriteLine("Введіть ID зображення, яке потрібно видалити:");
        int imageId = Convert.ToInt32(Console.ReadLine());

        string query = "DELETE FROM Images WHERE id = @id";

        using (SqliteCommand command = new SqliteCommand(query, connection))
        {
            command.Parameters.AddWithValue("@id", imageId);

            int rowsAffected = command.ExecuteNonQuery();
            Console.WriteLine($"{rowsAffected} рядків видалено.");
        }
    }
}

```

```

MSBuild version 17.9.8+b34f75857 for .NET
Determining projects to restore...
All projects are up-to-date for restore.
4.3 -> /Users/dmytro.soltusyuk/Work/university/Бази_даних/additional_3/bin/Debug/net8.0/4.3.dll

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:00.47
З'єднання з базою даних SQLite успішно встановлено.
Таблицю 'Images' успішно створено або вона вже існує.
Оберіть дію:
1. Завантажити зображення
2. Отримати список зображень
3. Видалити зображення
0. Вийти
1
Введіть повний шлях до зображення:
/Users/dmytro.soltusyuk/Downloads/engine.png
Введіть назву зображення:
Двигун
1 рядків додано.
Оберіть дію:
1. Завантажити зображення
2. Отримати список зображень
3. Видалити зображення
0. Вийти
2
Список зображень:
ID: 1, Назва: Двигун
Оберіть дію:
1. Завантажити зображення
2. Отримати список зображень
3. Видалити зображення
0. Вийти
3
Введіть ID зображення, яке потрібно видалити:
1
1 рядків видалено.
Оберіть дію:
1. Завантажити зображення
2. Отримати список зображень
3. Видалити зображення
0. Вийти
2
Список зображень:
Оберіть дію:
1. Завантажити зображення
2. Отримати список зображень
3. Видалити зображення
0. Вийти
0
0

```

## Висновок

Розроблено консольний додаток для роботи з зображеннями у базі даних SQLite. Програма забезпечує користувача зручним та ефективним інтерфейсом для завантаження, перегляду та видалення зображень. Вона може бути використана в різноманітних сценаріях, де потрібно працювати з великим обсягом зображень та забезпечити їхню ефективну організацію та керування.