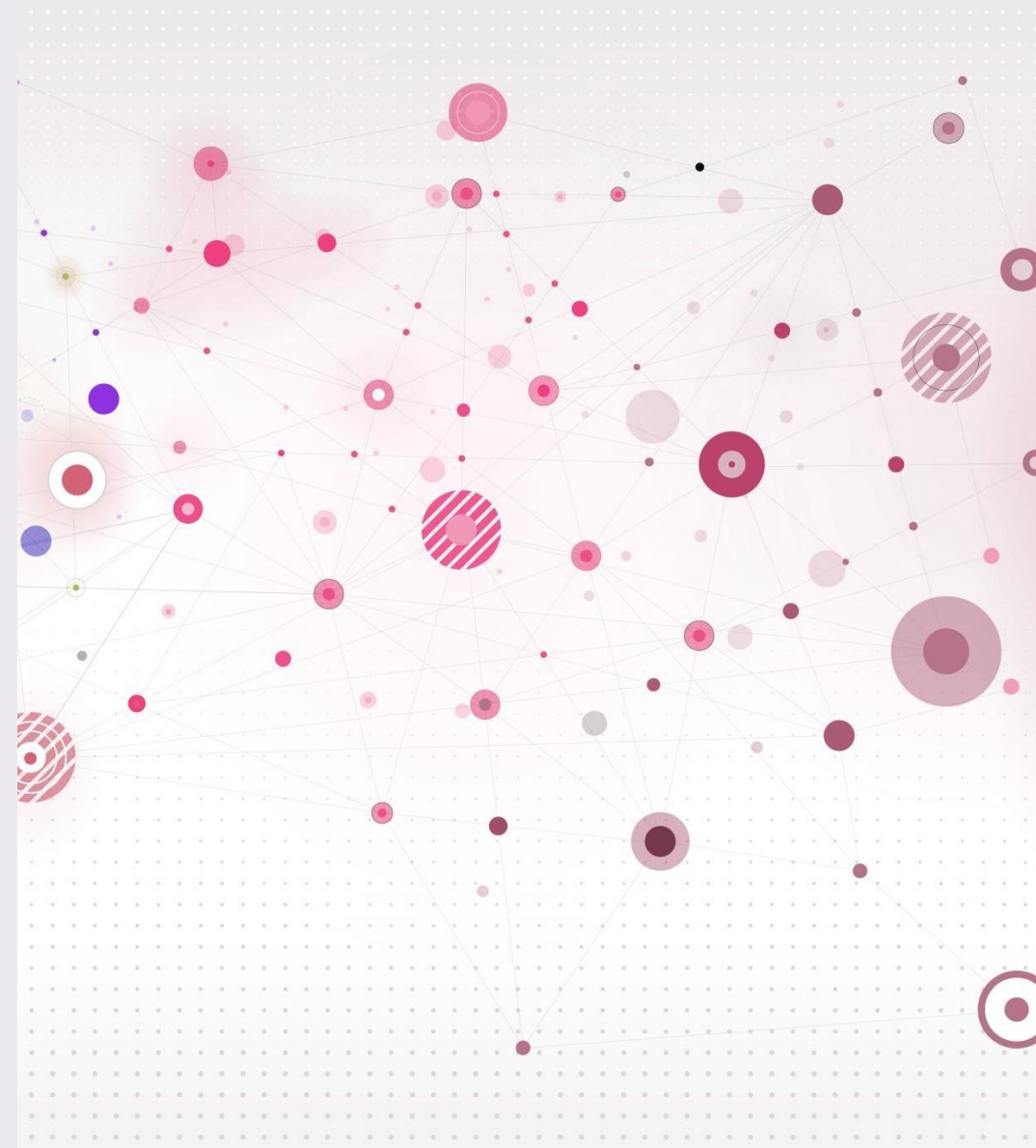


Додаток «Книжковий магазин»

Автор курсового проекту: Кізілпінар Світлана

Керівник проекту: Рубан Сергій Анатолійович



АНОТАЦІЯ

Темою даної курсової роботи була розробка бази даних "Книжковий магазин" та її реалізація за допомогою СУБД та програмних засобів Microsoft. Тому метою даної роботи є підвищення ефективності роботи програмного засобу для управління базою даних. Основними засобами для виконання цього завдання були програмні середовища SQL Server Management Studio 19 та Microsoft Visual Studio 2022.

Вступ

Предметною областю даної курсової роботи є проектування та розробка бази даних книжкового магазину. Система книгарні крім інформації про продажі включає в себе докладні відомості про товари даного магазину, а також клієнтів.

Про гостроту проблеми, що розглядається говорить той факт, що більшу частину свого часу адміністрація магазину витрачає на оформлення різної документації та звітів, а так само на пошук інформації про наявність в магазині деякої книги, журналу певного жанру або певного автора. Внесення всіх цих даних в єдину базу дозволить систематизувати їх, а також спростити співробітникам доступ до необхідної для роботи інформації, що в наслідок тягне за собою заощадження коштів і часу, потрібних для вилучення необхідних відомостей, що повинно позитивно позначитися на ефективності роботи магазину. Тому дана тема є актуальною і використання баз даних для спрощення роботи магазинів набуло широкого поширення в сучасному світі. Кінцевою метою розробки бази даних книгарні та автоматизованої системи для роботи з нею є підвищення якості обслуговування клієнтів і поліпшення колекції книг бібліотеки, магазину.

1. Постановка задачі

Необхідно зберігати таку інформацію про книги: назва книги, ПІБ автора, назва видавництва, кількість сторінок, жанр, рік видання, собівартість, ціна для продажу, чи є книга продовженням якоїсь іншої книги (наприклад, друга частина діалогії).

Додаток повинна дозволяти: додавати книги, видаляти книги, редагувати параметри книг, продавати книги, списувати книги, вносити книги в акції (наприклад, тиждень книг новорічної тематики зі знижкою 10%), відкладати книги для конкретного клієнта.

Додаток повинен надати функціональність пошуку книг за такими параметрів: назва книги, автор, жанр.

Додаток повинна надавати можливість переглянути список новинок, список найпопулярніших книг, список найпопулярніших авторів, список найпопулярніших жанрів за підсумками дня, тижня, місяця, року.

Необхідно передбачити можливість входу за логіном та паролем.

2. Основна частина

2.1 Теоретичні відомості про Visual Studio 2022

Visual Studio дозволяє програмістам створювати власні програми Win32/Win64, а також програми для .NET Framework розвивати. Крім того, за допомогою Visual Studio можна розробляти програми Windows , динамічні веб- сайти або веб-служби для Інтернету / інтрнету або служби Azure.

Visual Studio вважається комфортним середовищем розробки з багатьма функціями. Редактор підтримує напр. Б. онлайн-довідка, яка залежить від положення курсору , показ і приховування блоків коду, підсвічування ключових слів кольором, автоматична перевірка синтаксису та IntelliSense , автоматичне додавання методів і функцій під час введення вихідного тексту. Крім того, середовище розробки пропонує графічні інтерфейси для інтеграції веб- служб , бібліотек ActiveX і .NET, «Server Explorer» для доступу до джерел даних, таких як В. Microsoft SQL Server , служба повідомлень Windows або WMI , а також WYSIWYG-Редактори для розробки інтерфейсу користувача Windows і веб-додатків, редагування піктограм і XML-документів.

Visual Studio містить вбудований налагоджувач. Він містить функцію «Редагувати та продовжити» та дозволяє подальше підключення до процесів, які вже запущені, як на локальному комп'ютері, так і через мережу.

2.2 Теоретичні відомості про SQL Server Management Studio 19

Microsoft SQL Server Management Studio (SSMS) — це програма, розроблена корпорацією Майкрософт , інтегроване середовище для керування будь-якою інфраструктурою SQL. Використовується SSMS для доступу, налаштування, керування, адміністрування та розробки всіх компонентів SQL Server, бази даних SQL Azure , керованого екземпляра SQL Azure , SQL Server на Azure VM та Azure Synapse Analytics . SSMS надає єдину комплексну утиліту, яка об'єднує широку групу графічних інструментів із багатьма багатими редакторами сценаріїв, щоб забезпечити доступ до SQL Server для розробників і адміністраторів баз даних усіх рівнів кваліфікації.

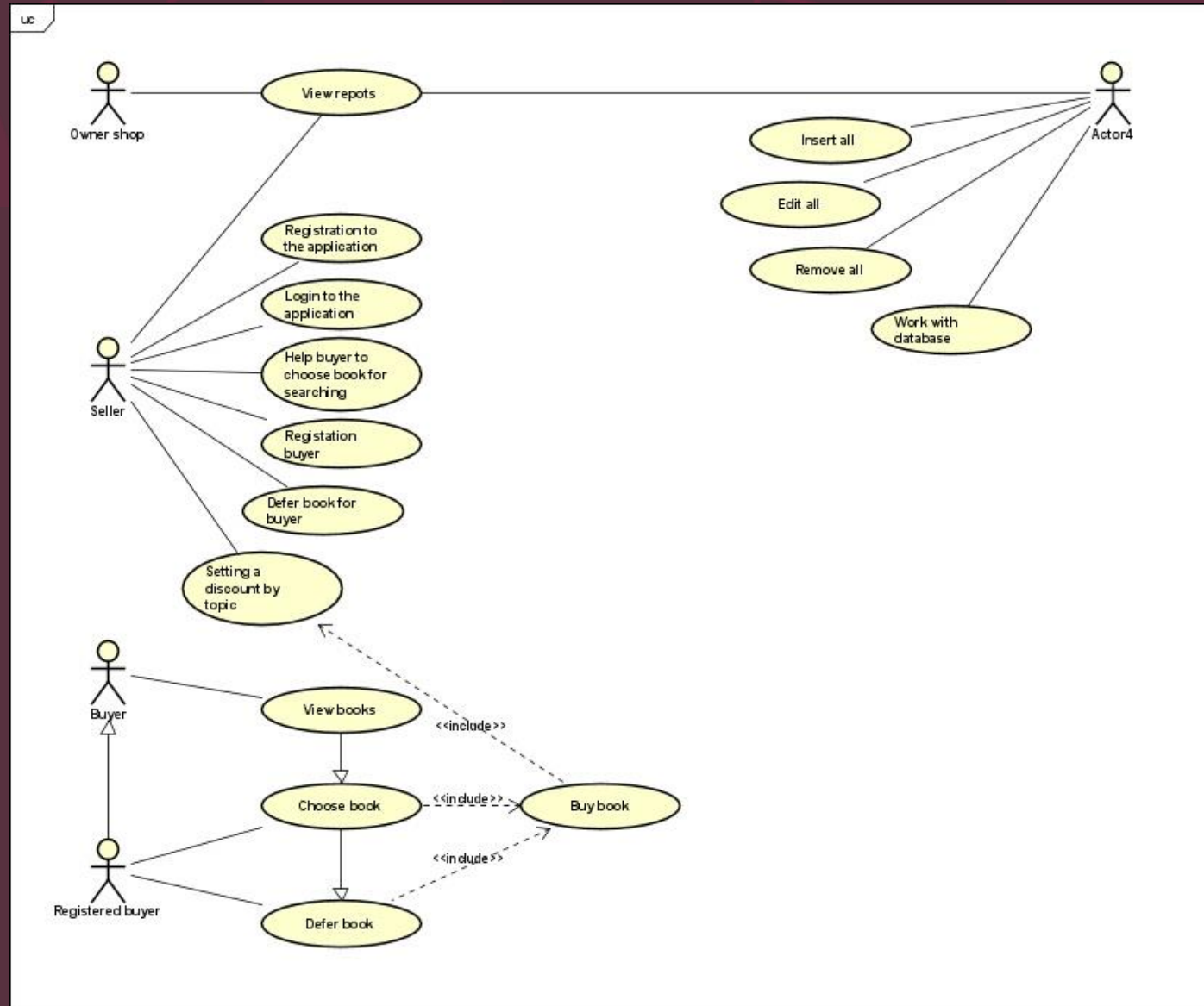
3. Практична частина

3.1 Проектування системи

Під час проектування інтерфейсу програми використовувалась платформа розробки Windows Forms, яка підтримує широкий набір функцій розробки програм, включаючи елементи керування, графіку, прив'язку даних і введення користувачами. Windows Forms містить візуальний конструктор Visual Studio з функцією перетягування, щоб легко створювати програми Windows Forms.

3.2 Проектування алгоритмів

Є дві дійові особи: user і admin. User має доступ до тих самих можливостей, що і admin за винятком редагування співробітників, додавання і видалення співробітників (мал. 3.1).



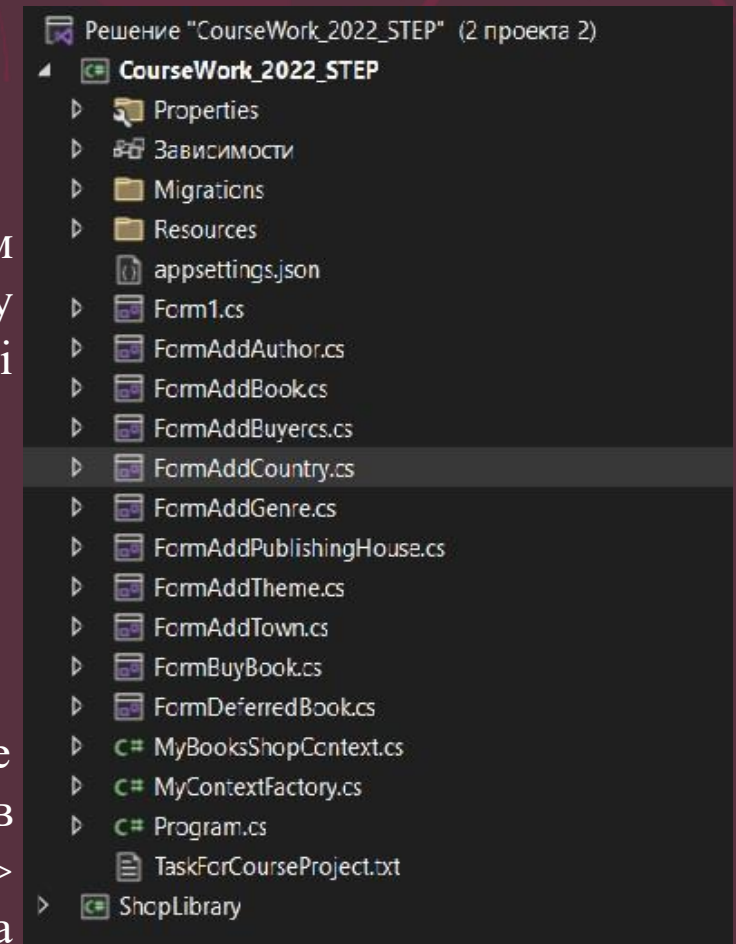
Мал. 3.1 Astah UseCase Diagramm

3.3 Проектування класів

В проекті використана методологія Code-First з використанням Migrations. Code First Migrations — це рекомендований спосіб розвитку схеми бази даних програми. Міграції надають набір інструментів, які дозволяють:

- Створіть початкову базу даних, яка працює з моделлю EF
- Створення міграцій для відстеження змін, які вносяться у модель EF
- Тримає базу даних в актуальному стані з урахуванням цих змін

Робочий процес розробки в підході, який базується на першому коді, буде таким: створення або зміна класи домену -> налаштування цих класів домену за допомогою Fluent-API або атрибутів анотації даних -> створення або оновлення схеми бази даних за допомогою міграції на основі коду.



Мал. 3.2

3.4 Проектування бази даних

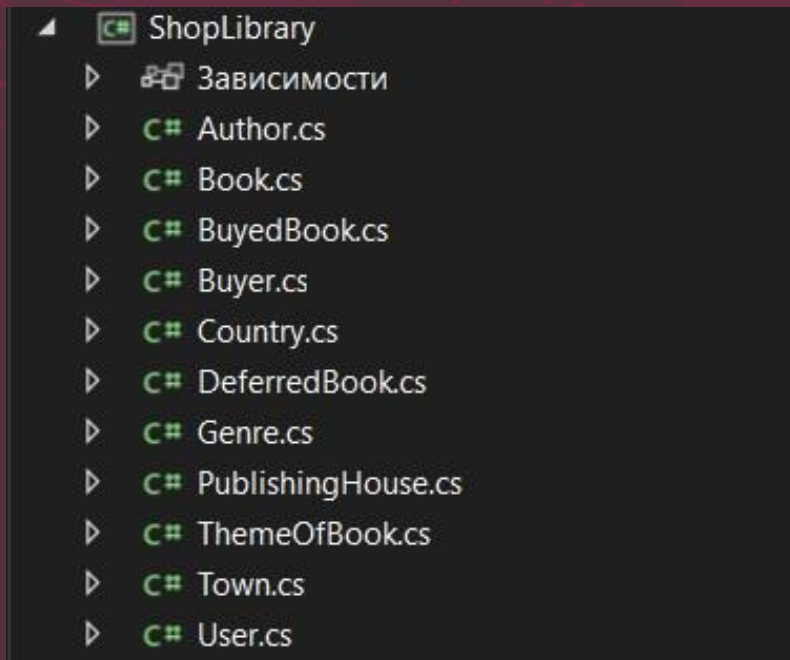
```
public class MyBooksShopContext : DbContext
{
    public DbSet<Book> Books { get; set; } = null!;
    public DbSet<Author> Authors { get; set; } = null!;
    public DbSet<Genre> Genres { get; set; } = null!;
    public DbSet<PublishingHouse> PublishingHouses { get; set; } = null!;
    public DbSet<Country> Countries { get; set; } = null!;
    public DbSet<Town> Towns { get; set; } = null!;

    public DbSet<BuyedBook> BuyedBooks { get; set; } = null!;
    public DbSet<Buyer> Buyers { get; set; } = null!;
    public DbSet<DeferredBook> DeferredBooks { get; set; } = null!;
    public DbSet<ThemeOfBook> ThemeOfBooks { get; set; } = null!;
    public DbSet<User> Users { get; set; } = null!;

    public MyBooksShopContext(DbContextOptions<MyBooksShopContext> options) : base(options) { }
}
```

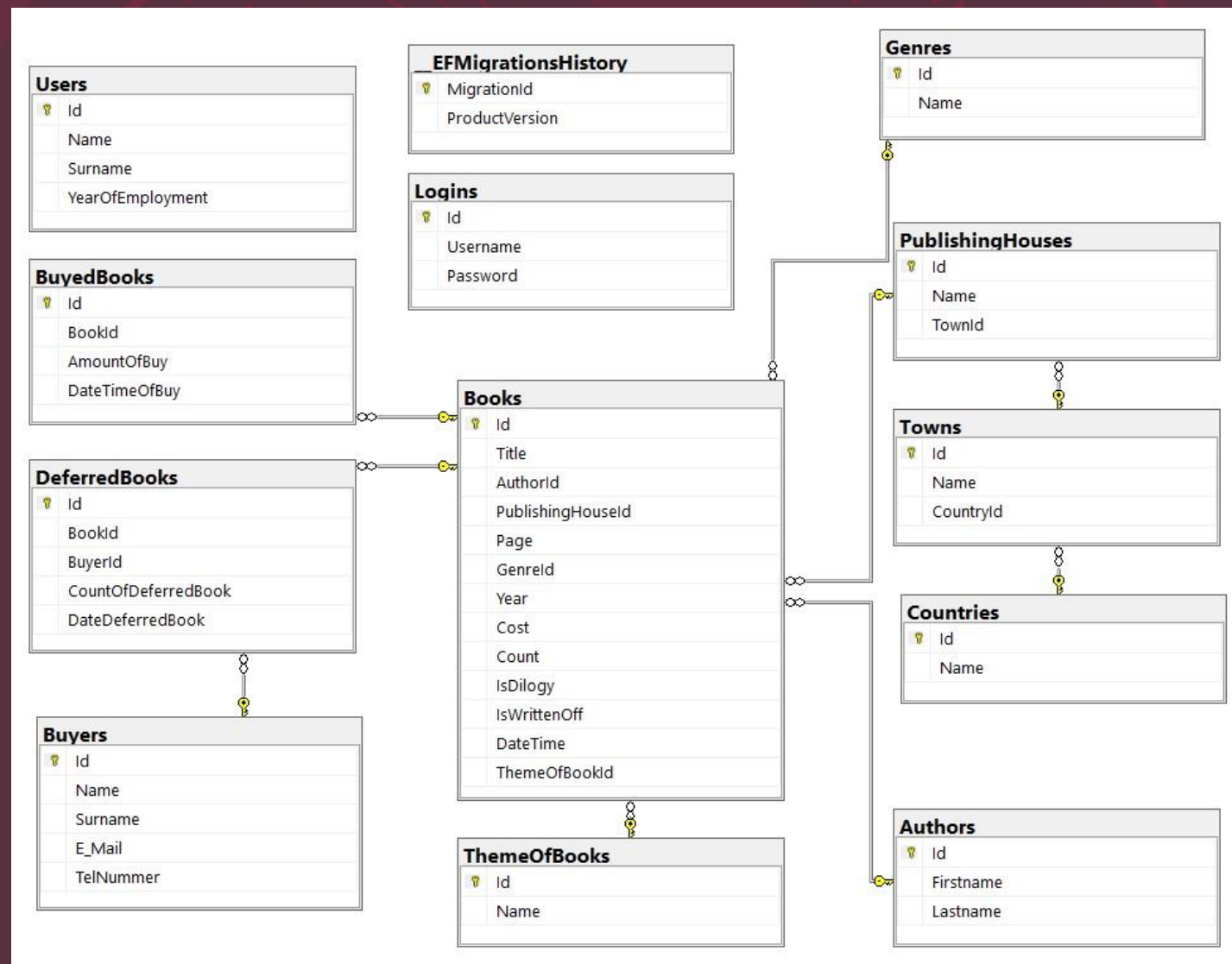
```
public class MyContextFactory : IDesignTimeDbContextFactory<MyBooksShopContext>
{
    public MyBooksShopContext CreateDbContext(string[] args)
    {
        var optionsBuilder = new DbContextOptionsBuilder<MyBooksShopContext>();
        IConfigurationBuilder builder = new ConfigurationBuilder(); //Microsoft.Extensions.Configuration NuGet
        builder.SetBasePath(Directory.GetCurrentDirectory()); //Microsoft.Extensions.Configuration.Json NuGet
        builder.AddJsonFile("appsettings.json"); //Microsoft.Extensions.Configuration.Json NuGet
        IConfigurationRoot config = builder.Build();
        string connStr = config.GetConnectionString("sqlConnStr");
        optionsBuilder.UseSqlServer(connStr); //Microsoft.EntityFrameworkCore.SqlServer NuGet
        var options = optionsBuilder.Options;
        return new MyBooksShopContext(options);
    }
}
```

```
{
  "exclude": [
    "**/bin",
    "**/bower_components",
    "**/jspm_packages",
    "**/node_modules",
    "**/obj",
    "**/platforms"
  ],
  "ConnectionStrings": {
    "sqlConnStr": "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=BooksShop_Migration;Integrated Security=True;"
  }
}
```



Мал. 3.3 Класи Entity

Мал. 3.4 SQL Database



3.4 Створення звітів

За допомогою звітів можна переглядати, форматовувати та підсумовувати дані. Наприклад, можна створити простий звіт, у якому будуть представлені номери телефонів усіх ваших контактів клієнтів, або зведений звіт за даними про збут за різні часові проміжки. Звіт складається з даних, які видобуваються з таблиць Database.

```
private void btnSearchByTitle_Click(object sender, EventArgs e)
{
    dataGridViewInfo.DataSource = null;
    using (MyBooksShopContext context = new MyBooksShopContext(options))
    {
        var book = context.Books.Where(t => t.Title == txtSearchName.Text)
            .Select(t => new
            {
                t.Id,
                t.Title,
                t.Author.Firstname,
                t.Author.Lastname,
                Publishing_House = t.PublishingHouse.Name,
                Pages = t.Page,
                Genre = t.Genre.Name,
                t.Year,
                t.Cost,
                t.Price,
                t.Count,
                t.DateTime,
                t.IsDilogy,
                t.IsWrittenOff
            }).ToList();
        if (book != null)
            dataGridViewInfo.DataSource = book;
        else
            MessageBox.Show("The book is not found.");
    }
}
```

```
private void btnSearchByAuthor_Click(object sender, EventArgs e)
{
    dataGridViewInfo.DataSource = null;
    using (MyBooksShopContext context = new MyBooksShopContext(options))
    {
        var book = context.Books.Where(t => t.Author.Firstname == txtSearchName.Text
            && t.Author.Lastname == txtSearchSurname.Text)
            .Select(t => new
            {
                t.Id,
                t.Title,
                t.Author.Firstname,
                t.Author.Lastname,
                Publishing_House = t.PublishingHouse.Name,
                Pages = t.Page,
                Genre = t.Genre.Name,
                t.Year,
                t.Cost,
                t.Price,
                t.Count,
                t.DateTime,
                t.IsDilogy,
                t.IsWrittenOff
            }).ToList();
        if (book != null)
            dataGridViewInfo.DataSource = book;
        else
            MessageBox.Show("The book is not found.");
    }
}
```

Список книг, що найбільше продаються:

```
private void btnTopBuedBook_Click(object sender, EventArgs e)
{
    using (MyBooksShopContext context = new MyBooksShopContext(options))
    {
        dataGridViewInfo.DataSource = context.BuyedBooks
            .GroupBy(t => t.Book.Title).Select(g => new { Title = g.Key, Count = g.Sum(x => x.AmountOfBuy) })
            .OrderByDescending(g=>g.Count).ToList();
    }
}
```


Список найпопулярніших авторів:

```
private void btnPopularAuthors_Click(object sender, EventArgs e)
{
    using (MyBooksShopContext context = new MyBooksShopContext(options))
    {
        dataGridViewInfo.DataSource = context.BuyedBooks
            .GroupBy(t => t.Book.Author.Firstname + " " + t.Book.Author.Lastname).Select(g => new { Author = g.Key,
Count = g.Sum(x => x.AmountOfBuy) })
            .OrderByDescending(g => g.Count).ToList();
    }
}
```

Список найпопулярніших жанрів за підсумками дня:

```
private void btnPopularGenresOfDay_Click(object sender, EventArgs e)
{
    using (MyBooksShopContext context = new MyBooksShopContext(options))
    {
        dataGridViewInfo.DataSource = context.BuyedBooks.Where(t => t.DateTimeOfBuy >= DateTime.Today).Select(t
=> new { Genre = t.Book.Genre.Name, Amount = t.AmountOfBuy, Date = t.DateTimeOfBuy })
            .GroupBy(t=>t.Genre).Select(g=> new { NameOfGenre = g.Key, Count = g.Sum(x=>x.Amount) })
            .OrderByDescending(g=>g.Count).ToList();
    }
}
```

Список найпопулярніших жанрів за підсумками дня:

```
private void btnPopularGenresOfWeek_Click(object sender, EventArgs e)

{

    using (MyBooksShopContext context = new MyBooksShopContext(options))

    {

        dataGridViewInfo.DataSource = context.BuyedBooks.Where(t => t.DateTimeOfBuy >=
DateTime.Today.AddDays(-7)).Select(t => new { Genre = t.Book.Genre.Name, Amount = t.AmountOfBuy, Date =
t.DateTimeOfBuy })

        .GroupBy(t => t.Genre).Select(g => new { NameOfGenre = g.Key, Count = g.Sum(x => x.Amount) })

        .OrderByDescending(g => g.Count).ToList();

    }

}
```

4. Висновки

За останні десятиліття бази даних стали основою інформаційних систем і докорінно змінили методи роботи багатьох організацій. Розвиток технології баз даних призвело до створення досить потужних і зручних в експлуатації програм. Вони являють собою зручний і надійний спосіб зберігання номенклатурних даних продукції, відомостей про клієнтські бази та іншої інформації, чиї обсяги перевищують можливості обліку за допомогою звичайних таблиць.

Ця програма як найкраще є доказом то, що систематизація допоможе зберігати данні, отримувати данні, необхідні для звітності. Є гнучкою завдяки Code First Migrations, можна як додавати, так й прибирати данні чи корегувати їх.

Використання даної програми, на боці користувача, не зазначає мати спеціальних навичок.

Books shop

×

BooksAuthorsGenresPublishing housesTownsCountriesTheme of bookBuyed booksBuyersDeferred books

Log In

Add database

Show all data

Work with actions

Theme of books:
New year

Amount of discount:
0

Work with book

Show new books

Write off book

Sale book

Defer book for client

Work with client

Add new buyer

Edit buyer

Delete buyer

Work with Database

Add book

Add author

Add genre

Add publishing house

Add town

Add country

Add theme

Edit book

Edit author

Edit genre

Edit publishing house

Edit town

Edit country

Edit theme

Delete book

Delete author

Delete genres

Delete publishing

Delete town

Delete country

Delete theme

Search book

Name/Title

Surname of author

by titleby authorby genre

Show TOP of buyed books

Top buyed books

Popular authors

Popular genres of day

Popular genres of week

Popular genres of month

Popular genres of year

Мал. 3.7 Головне меню

Add book

Title

Author

Publishing house

Pages

Genre

Year

Theme

Cost

Count

Date input

Is dilogy?
☐ True
☐ False

Is written off?
☐ True
☐ False

OK Cancel

Add Buyer

Firstname

Lastname

E-Mail

Telephone number

OK Cancel

Add Town

Name

Country

OK Cancel

Add Country

Name

OK Cancel

Add Publishing House

Name

Town

OK Cancel

Add Genre

Name

OK Cancel

Add Theme

Name of theme

OK Cancel

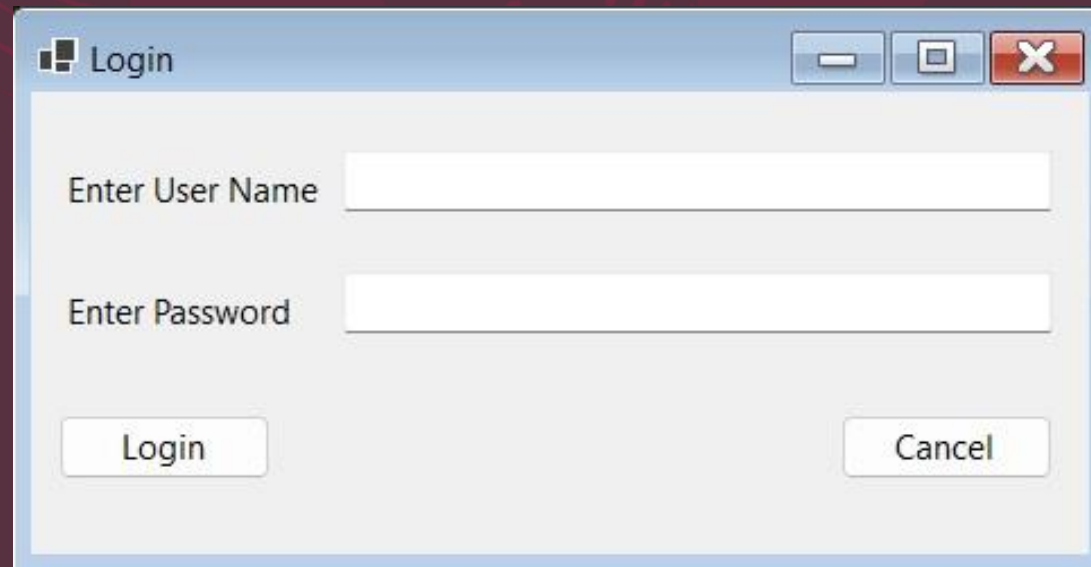
Add Author

Name

Surname

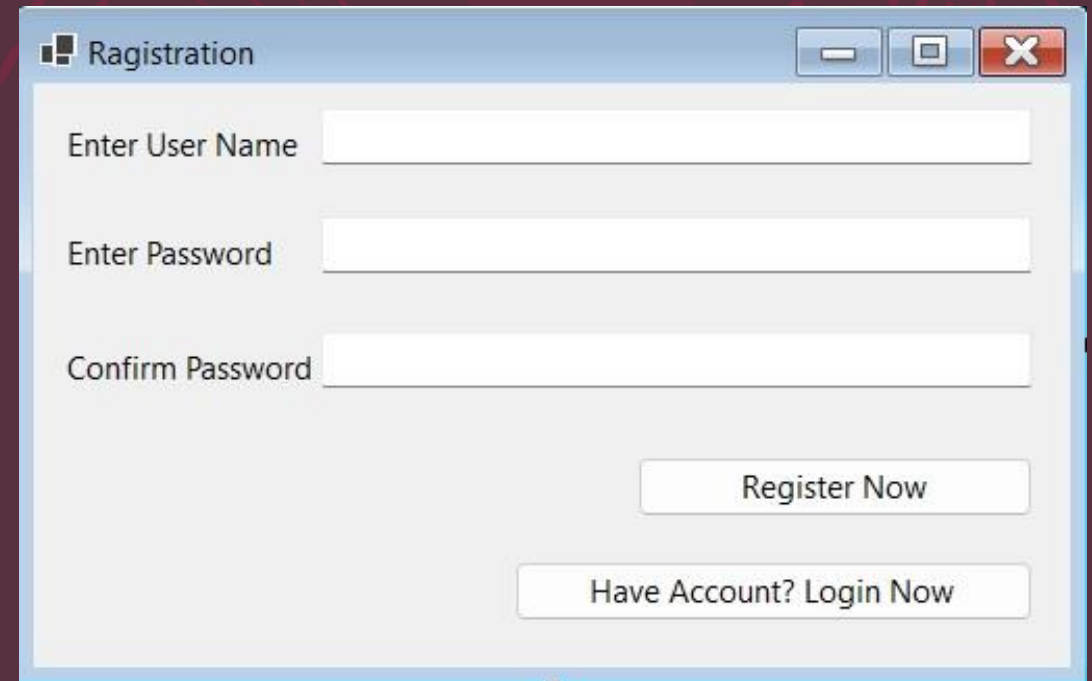
OK Cancel

Ма.л. 3.8 Add book, Add Buyer, Add Country, Add Genre, Add Publish House, Add Theme, Add Town, Add Author



A Windows-style dialog box titled "Login". It features a title bar with standard minimize, maximize, and close buttons. The main area contains two text input fields: "Enter User Name" and "Enter Password". Below these fields are two buttons: "Login" on the left and "Cancel" on the right.

Login



A Windows-style dialog box titled "Registration". It features a title bar with standard minimize, maximize, and close buttons. The main area contains three text input fields: "Enter User Name", "Enter Password", and "Confirm Password". Below these fields are two buttons: "Register Now" and "Have Account? Login Now".

Registration

Мал. 3.9 Forms 'Login' and 'Registration'