

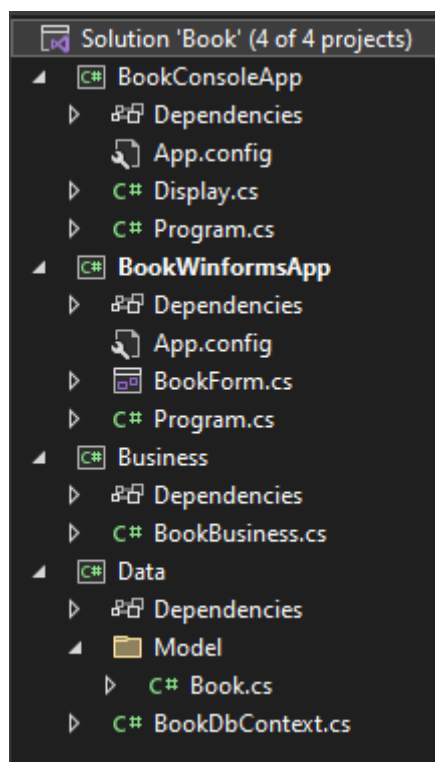
Описание на проекта

Проектът "Books" представлява система за управление на книги. Проектът е организиран по трислоен модел - слой за данни, слой за услуги и презентационен слой.

Включва Windows Forms приложение, конзолно приложение и класове за данни и бизнес логика, които позволяват на потребителите да извършват операции като добавяне, редактиране, преглеждане и изтриване на книги в база данни.

Използвани са технологиите C#, Entity Framework, .NET Console, Windows Forms, .NET Framework, LINQ.

Структура на проекта



Конзолно приложение (BookConsoleApp):

Предоставя интерактивен текстов интерфейс за операции с книги.

Използва Console за обработка на вход и изход от потребителя.

Windows Forms приложение (BookWinformsApp):

Предоставя графичен потребителски интерфейс за операции с книги на десктоп компютъра.

Използва Windows Forms за създаване на графичния интерфейс.

Class Library - Business:

Слой за услуги (бизнес логика), който съдържа бизнес логика и функционалност за извършване на операции с книги.

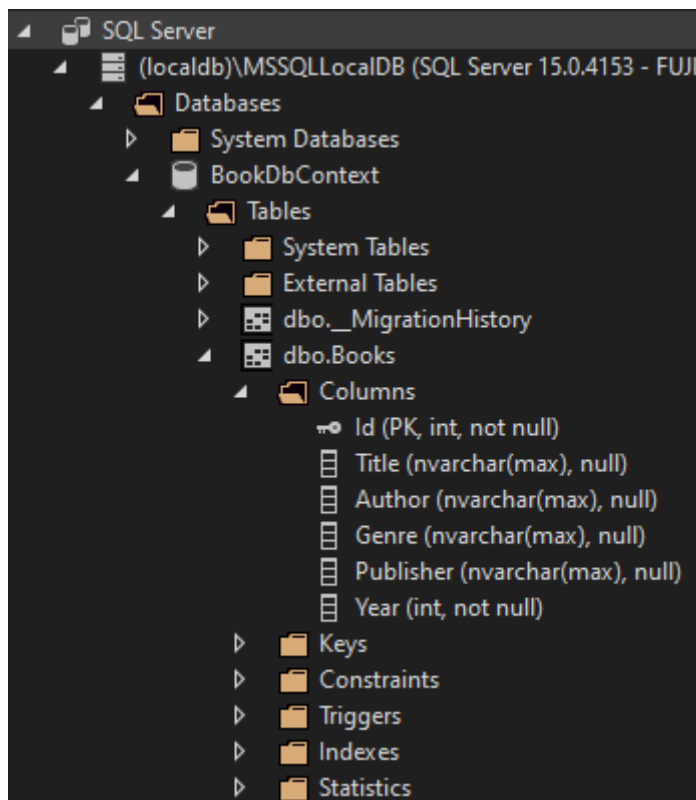
Включва валидация на данни, манипулиране на данни и връзка между интерфейса и слоя за данни.

Class Library - Data:

Слой за данни, който включва класове за работа с базата данни и моделите на данни.

Използва Entity Framework за взаимодействие с базата данни.

Създаване на базата данни



При изграждането на базата данни на приложението е използван Code First подход- първо се създават моделите на данни, а базата данни се генерира автоматично от тези модели. Този подход позволява по-лесна и гъвкава

работа с моделите на данни, като разработчиците имат пълен контрол върху тях.

Методът за създаване на базата данни включва използването на Entity Framework за манипулиране на базата данни.

Модел на данните (Data Model)

Първо създаваме модел на данните, който описва структурата на данните, които искаме да съхраняваме в базата данни. В нашия случай, моделът е класът Book, който съдържа информация за книгите, включително заглавие, автор, жанр, издател и година на издаване.

Създаване на контекст на базата данни (Database Context)

Следващата стъпка е създаването на контекст на базата данни. Този контекст управлява връзката с базата данни и включва DbSet за моделите на данните. В нашия случай, контекстът на базата данни е класът BookDbContext, който наследява DbContext и включва DbSet за книгите.

Конфигуриране на контекста на базата данни

В контекста на базата данни се конфигурира връзката с базата данни и начинът на създаване на базата данни. В нашия случай, използваме подхода Code First с Database.SetInitializer(new CreateDatabaseIfNotExists<BookDbContext>()), което означава, че базата данни ще бъде автоматично създадена, ако не съществува.

При стартиране на приложението, ако базата данни все още не съществува, се извиква конструкторът на BookDbContext, който настройва връзката с базата данни и активира автоматичното създаване на базата данни, ако тя не съществува.

База данни

В резултат на горепосочените стъпки, ще бъде създадена база данни, която съдържа таблица с информация за книгите. Всяка колона в тази таблица отговаря на свойство от класа Book. За връзка с базата данни, проектът използва конфигурационния файл App.config, където са посочени настройките за връзка, включително нейното име и връзката с нея.

Описание на базата данни

Базата данни се казва "BookDbContext" и включва една таблица (Books) за съхранение на данни за книги. Таблицата има следните колони:

Id: Уникален идентификатор на книгата (Primary Key. Int, not null).

Title: Заглавие на книгата (nvarchar).

Author: Автор на книгата (nvarchar).

Genre: Жанр на книгата (nvarchar).

Publisher: Издател на книгата (nvarchar).

Year: Година на издаване на книгата (int).

Таблицата се създава автоматично от Entity Framework при първоначалното създаване на базата данни. Освен това, ако при стартиране на приложението вече съществува база данни "BookDbContext", няма да бъде създадена нова, а данните ще бъдат запазени в съществуващата база данни.

Слой за данни (Data Layer)

Служи за автоматично създаване на базата данни. При първоначално стартиране на приложението осигурява автоматично създаване на базата данни, ако тя не съществува, чрез `Database.SetInitializer(new CreateDatabaseIfNotExists<BookDbContext>())`.

Клас Book

Класът Book представлява модел на книга в базата данни. Той съдържа следните свойства:

Id: Уникален идентификатор на книгата (int).

Title: Заглавие на книгата (string).

Author: Автор на книгата (string).

Genre: Жанр на книгата (string).

Publisher: Издател на книгата (string).

Year: Година на издаване на книгата (int).

Клас BookDbContext

Класът `BookDbContext` е контекст на базата данни, който управлява връзката с базата данни и включва `DbSet` за класа `Book`. Той съдържа следните свойства и методи:

- `Books`: `DbSet` за книгите в базата данни.
- `BookDbContext()`: Конструкторът на контекста на базата данни. Инициализира контекста на базата данни и включва автоматично създаване на базата данни, ако тя не съществува. Също така вмъква първоначални данни, ако базата данни е нова. Осигурява връзка с базата данни и позволява съхраняване на данни за книгите.
- `SeedData()`: Метод за вмъкване на първоначални данни в базата данни. Вмъква пет популярни книги в базата данни при инициализацията ѝ, ако тя е нова. Това се извършва единствено ако базата данни е нова и не съдържа вече данни.

Слой за услуги (бизнес слой)

Слой за услуги (бизнес слой) в приложението "Books" е отговорен за предоставянето на бизнес логика и функционалности върху данните за книгите. Този слой управлява взаимодействието между слоя за данни и презентационния слой, като осигурява обработка на данните и изпълнява работните правила на приложението.

Използва `Entity Framework` за взаимодействие с базата данни чрез `ORM (Object-Relational Mapping)` и `LINQ (Language Integrated Query)` за заявки към базата данни за извличане на данни.

Клас `BookBusiness`

Това е основният клас в бизнес слоя. Той съдържа методи за извличане, добавяне, редактиране и изтриване на данни за книгите.

Функционалности и методи

- `GetAll()`: Методът връща списък с всички книги от базата данни.
- `Get(int id)`: Методът връща книга по даден уникален идентификатор.
- `Add(Book book)`: Методът добавя нова книга към базата данни.
- `Update(Book book)`: Методът актуализира съществуваща книга в базата данни.

- Delete(int id): Методът изтрива книга от базата данни по даден уникален идентификатор.

Работа с контекст на базата данни

Всеки от методите използва контекст на базата данни (BookDbContext) за взаимодействие с базата данни.

Използва се using конструкция, за да се гарантира, че контекстът на базата данни се освобождава коректно след използване на методите.

Презентационен слой

Проектът има два начина за реализация на презентационния слой:

- Конзолно приложение: Използва се текстов интерфейс чрез конзолата (команден ред) за визуализация на данните и взаимодействие с потребителя посредством команди.
- Windows Forms приложение: Използва се графичен потребителски интерфейс с помощта на Windows Forms за визуализация на данните и интерактивно взаимодействие с потребителя.

Конзолно приложение

Презентационният слой в конзолното приложение "Books" е отговорен за визуализиране на потребителския интерфейс и взаимодействието с потребителя. Този слой предоставя конзолен команден ред, чрез който потребителят може да извършва операции върху данните за книгите.

Използва слоя за услуги за достъп до бизнес логиката и функционалностите за обработка на данните.

Презентационният слой комуникира с бизнес слоя чрез обект на класа BookBusiness. Извикват се методите от бизнес слоя за извличане, добавяне, редактиране и изтриване на данни за книгите.

На екрана на потребителя се извежда меню с наличните опции за операции с книги. Потребителят избира желаната операция, като въвежда нейния номер. Очаква се да въведе коректни данни, като година, заглави на книга и други информационни полета за добавяне или промяна на книгата.

След изпълнението на операцията се връща обратно към менюто за нов избор или завършване на приложението.

Презентационният слой е реализиран чрез подхода при който потребителят предоставя входни данни, които се обработват от приложението, а резултатът се връща обратно на потребителя.

Класове

- Program: Главният клас на приложението, който съдържа метода Main() в който се създава обект от клас Display.
- Display: Класът, който представлява презентационния слой. Той съдържа методи за взаимодействие с потребителя и извикване на методи от бизнес слоя.

Методи в клас Display

- Run(): Методът, който се изпълнява за стартиране на приложението и включва основната логика за взаимодействие с потребителя.
- Input(string message): Метод за въвеждане на команда от потребителя.
- ShowAllBooks(): Метод за показване на списък с всички книги в базата данни.
- GetBookById(BookBusiness bookBusiness): Метод за показване на информация за книга по даден идентификатор.
- AddNewBook(BookBusiness bookBusiness): Метод за добавяне на нова книга в базата данни.
- UpdateBook(BookBusiness bookBusiness): Метод за промяна на информация за съществуваща книга в базата данни.
- DeleteBook(BookBusiness bookBusiness): Метод за изтриване на книга от базата данни.

WindowsForm приложение "BookForm"

Презентационният слой на това приложение е базиран на WindowsForm, което предоставя графичен потребителски интерфейс за взаимодействие със системата за управление на книги. Този слой позволява на потребителя да визуализира, добавя, редактира и изтрива записи за книги в базата данни.

Клас BookForm: Главният клас на WindowsForm приложението, който представлява главния прозорец на приложението. Визуализира списък с книги в DataGridView компонент и предоставя TextBox полета за въвеждане на данни за книга и NumericUpDown поле за година на издаване. Осигурява бутони за добавяне, редактиране и изтриване на книги.

Използвани методи:

- RefreshDataGridView(): Методът се използва за обновяване на данните в DataGridView компонента, като извлича списък с всички книги от бизнес слоя и ги зарежда в DataGridView.
- buttonAdd_Click(): Методът се извиква при натискане на бутона за добавяне на нова книга. Създава обект от клас Book със зададените данни от текстовите полета и годината от числовия компонент и го добавя към базата данни.
- buttonUpdate_Click(): Методът се извиква при натискане на бутона за редактиране на книга. Извлича избраната книга от DataGridView, създава обект с новите данни и ги обновява в базата данни.
- buttonDelete_Click(): Методът се извиква при натискане на бутона за изтриване на книга. Извлича избраната книга от DataGridView и я изтрива от базата данни.
- dataGridViewBooks_SelectionChanged(): Методът се извиква при промяна на избора в DataGridView и зарежда данните от избрания ред в текстовите полета за редактиране.