**Документация к работе по теме: “Разработка простой системы управления библиотекой”**

/// <summary>

**///** **Основной класс программы, содержащий точку входа в приложение.**

/// </summary>

public class Program

{

/// <summary>

**/// Метод, который является точкой входа в приложение.**

/// </summary>

/// <param name="args">Массив аргументов командной строки.</param>

static void Main(string[] args)

{

**// Создание экземпляров менеджеров книг и читателей**

BookManager bookManager = new BookManager();

ReaderManager readerManager = new ReaderManager();

// Основной цикл приложения

while (true)

{

**// Отображение доступных действий для пользователя**

Console.WriteLine("\nВыберите действие:");

Console.WriteLine("1. Добавить книгу");

Console.WriteLine("2. Удалить книгу");

Console.WriteLine("3. Просмотреть все книги");

Console.WriteLine("4. Добавить читателя");

Console.WriteLine("5. Удалить читателя");

Console.WriteLine("6. Просмотреть всех читателей");

Console.WriteLine("0. Выход");

var choice = Console.ReadLine();

**// Обработка выбора пользователя**

try

{

switch (choice)

{

case "1":

**// Добавление новой книги**

Console.Write("Введите идентификатор книги: ");

var bookId = Console.ReadLine();

Console.Write("Введите название книги: ");

var bookTitle = Console.ReadLine();

Console.Write("Введите автора книги: ");

var bookAuthor = Console.ReadLine();

Console.Write("Введите год издания: ");

var yearPublishedInput = Console.ReadLine();

**// Проверка корректности года издания**

if (!int.TryParse(yearPublishedInput, out var yearPublished))

{

throw new Exception("Год издания должен быть числом.");

}

var book = new Book(bookId, bookTitle, bookAuthor, yearPublished);

bookManager.AddBook(book);

Console.WriteLine("Книга добавлена.");

break;

case "2":

**// Удаление книги**

Console.Write("Введите идентификатор книги для удаления: ");

var removeBookId = Console.ReadLine();

bookManager.RemoveBook(removeBookId);

Console.WriteLine("Книга удалена.");

break;

case "3":

**// Просмотр всех книг**

Console.WriteLine("Список всех книг:");

var allBooks = bookManager.GetAllBooks();

foreach (var b in allBooks)

{

Console.WriteLine($"ID: {b.Id}, Название: {b.Title}, Автор: {b.Author}, Год: {b.YearPublished}");

}

break;

case "4":

**// Добавление нового читателя**

Console.Write("Введите идентификатор читателя: ");

var readerId = Console.ReadLine();

Console.Write("Введите имя читателя: ");

var readerName = Console.ReadLine();

Console.Write("Введите email читателя: ");

var readerEmail = Console.ReadLine();

var reader = new Reader(readerId, readerName, readerEmail);

readerManager.AddReader(reader);

Console.WriteLine("Читатель добавлен.");

break;

case "5":

**// Удаление читателя**

Console.Write("Введите идентификатор читателя для удаления: ");

var removeReaderId = Console.ReadLine();

readerManager.RemoveReader(removeReaderId);

Console.WriteLine("Читатель удален.");

break;

case "6":

**// Просмотр всех читателей**

Console.WriteLine("Список всех читателей:");

var allReaders = readerManager.GetAllReaders();

foreach (var r in allReaders)

{

Console.WriteLine($"ID: {r.Id}, Имя: {r.Name}, Email: {r.Email}");

}

break

case "0":

**// Выход из приложения**

return;

default:

Console.WriteLine("Некорректный выбор, попробуйте еще раз.");

break;

}

}

catch (Exception ex)

{

**// Обработка исключений**

Console.WriteLine($"Ошибка: {ex.Message}");

}

}

}

}