using System;

using System.Collections.Generic;

public class Book

{

public string Title { get; set; }

public string Author { get; set; }

public int Year { get; set; }

public string ISBN { get; set; }

public override string ToString()

{

return $"Title: {Title}, Author: {Author}, Year: {Year}, ISBN: {ISBN}";

}

}

public class BookManager

{

private List<Book> books = new List<Book>();

public void AddBook(Book book)

{

books.Add(book);

}

public Book FindBookByTitle(string title)

{

return books.Find(b => b.Title.Equals(title, StringComparison.OrdinalIgnoreCase));

}

public void UpdateBook(Book book)

{

Book existingBook = FindBookByTitle(book.Title);

if (existingBook != null)

{

existingBook.Author = book.Author;

existingBook.Year = book.Year;

existingBook.ISBN = book.ISBN;

}

}

public void DeleteBook(string title)

{

Book book = FindBookByTitle(title);

if (book != null)

{

books.Remove(book);

}

}

public List<Book> GetAllBooks()

{

return books;

}

}

public class Program

{

private static BookManager bookManager = new BookManager();

public static void Main()

{

while (true)

{

Console.WriteLine("1. Add Book");

Console.WriteLine("2. View All Books");

Console.WriteLine("3. Search Book by Title");

Console.WriteLine("4. Update Book");

Console.WriteLine("5. Delete Book");

Console.WriteLine("6. Exit");

Console.Write("Enter your choice: ");

string choice = Console.ReadLine();

switch (choice)

{

case "1":

AddBook();

break;

case "2":

ViewAllBooks();

break;

case "3":

SearchBookByTitle();

break;

case "4":

UpdateBook();

break;

case "5":

DeleteBook();

break;

case "6":

Console.WriteLine("Exiting program...");

return;

default:

Console.WriteLine("Invalid choice. Please try again.");

break;

}

Console.WriteLine();

}

}

private static void AddBook()

{

Console.Write("Enter book title: ");

string title = Console.ReadLine();

Console.Write("Enter book author: ");

string author = Console.ReadLine();

Console.Write("Enter publication year: ");

int year = int.Parse(Console.ReadLine());

Console.Write("Enter ISBN: ");

string isbn = Console.ReadLine();

Book book = new Book { Title = title, Author = author, Year = year, ISBN = isbn };

bookManager.AddBook(book);

Console.WriteLine("Book added successfully!");

}

private static void ViewAllBooks()

{

List<Book> books = bookManager.GetAllBooks();

Console.WriteLine("All Books:");

foreach (var book in books)

{

Console.WriteLine(book);

}

}

private static void SearchBookByTitle()

{

Console.Write("Enter book title to search: ");

string title = Console.ReadLine();

Book book = bookManager.FindBookByTitle(title);

if (book != null)

{

Console.WriteLine("Book found:");

Console.WriteLine(book);

}

else

{

Console.WriteLine("Book not found.");

}

}

private static void UpdateBook()

{

Console.Write("Enter book title to update: ");

string title = Console.ReadLine();

Book existingBook = bookManager.FindBookByTitle(title);

if (existingBook != null)

{

Console.Write("Enter new author: ");

string author = Console.ReadLine();

Console.Write("Enter new publication year: ");

int year = int.Parse(Console.ReadLine());

Console.Write("Enter new ISBN: ");

string isbn = Console.ReadLine();

Book updatedBook = new Book { Title = title, Author = author, Year = year, ISBN = isbn };

bookManager.UpdateBook(updatedBook);

Console.WriteLine("Book updated successfully!");

}

else

{

Console.WriteLine("Book not found.");

}

}

private static void DeleteBook()

{

Console.Write("Enter book title to delete: ");

string title = Console.ReadLine();

bookManager.DeleteBook(title);

Console.WriteLine("Book deleted successfully!");

}

}