Library Management System in Python

Objective:

To develop a simple library management system that demonstrates your understanding of Python concepts such as variables, data types, operators, string manipulation, user input, conditional statements, loops, functions, modules, file handling, error handling, and object-oriented programming.

Task Overview:

You are tasked with creating a Python script named `library_management.py` that performs various functions to manage a collection of books in a library. This task will help you apply fundamental Python programming concepts in a practical project.

Detailed Task Instructions:

1. Class Definition

You will define two classes, `Book` and `Library`. The `Book` class will represent individual book objects, while the `Library` class will manage a collection of books.

- Book Class: Define a class `Book` with attributes like `title`, `author`, `year`, and `isbn` (isbn is book serial number you can use 1,2,3… etc).
  - Implement an `__init__` method to initialize these attributes.
  - Implement a `display` method to return a formatted string with the book's details.
- Library Class: Define a class `Library` to manage a collection of `Book` objects.
  - Implement an `__init__` method to initialize an empty list of books.
  - Implement methods to add, remove, find, and display books.
  - Implement methods to save and load the library collection to/from a file.

2. Book Class

The `Book` class will have attributes for `title`, `author`, `year`, and `isbn`. It will also include a `display` method to return a formatted string with the book's details

The `Book` class should include:
- An `__init__` method to set the title, author, year, and ISBN of the book.
- A `display` method to print the book's details in a readable format.

3. Library Class

The `Library` class will manage a list of books. It will have methods to add, remove, find, and display books, as well as to save and load the collection to/from a file.

The `Library` class should include:
- An `__init__` method to initialize an empty list to store books.
- An `add_book` method to add a book to the library.
- A `remove_book` method to remove a book from the library by its ISBN.
- A `find_book` method to find and display a book by its ISBN.
- A `display_books` method to display all books in the library.
- A `save_to_file` method to save the library's book list to a file. (file text format Title,Author,Year,ISBN for each book record per line)
- A `load_from_file` method to load the library's book list from a file. (file text format Title,Author,Year,ISBN for each book record per line)

4. User Interface

The main function will provide a menu-driven interface that allows users to interact with the library system. Users can add, remove, find, display, save, and load books through input prompts.

Implement a text-based user interface in the `main` function to allow users to interact with the library system.
The interface should include options to add, remove, find, display, save, and load books.
Use input prompts to get user data and perform the corresponding library operations.

Evaluation Criteria:

- Correctness and functionality of the script.
- Proper use of Python concepts and techniques.
- Code readability and documentation.
- Efficiency and creativity in solving the tasks.