**Задача: Управление базой данных книг с использованием пакетов readline-sync и fs**

Вам необходимо разработать программу для управления базой данных книг с использованием пакетов **readline-sync** и **fs** в Node.js. Программа должна предоставлять следующие функции: добавление книги, получение списка всех книг, обновление информации о книге и удаление книги.

1. Создайте новый проект и установите пакеты **readline-sync** и **fs** с помощью **npm**.
2. Создайте файл **database.js**, который будет содержать функции для работы с базой данных книг.
3. Импортируйте модули **readline-sync** и **fs** в файл **database.js**.
4. Создайте функцию **addBook()**, которая будет принимать данные о книге (title, description, author, year, price) и добавлять новую книгу в базу данных. Для этого считайте текущее содержимое файла базы данных с помощью **fs.readFileSync()**, преобразуйте его в надлежащий формат JSON, добавьте новую книгу в массив объектов, а затем запишите обновленные данные обратно в файл базы данных с помощью **fs.writeFileSync()**.
5. Создайте функцию **getAllBooks()**, которая будет возвращать список всех книг из базы данных. Считайте содержимое файла базы данных с помощью **fs.readFileSync()** и преобразуйте его в надлежащий формат JSON. Затем верните массив книг из объекта.
6. Создайте функцию **updateBook()**, которая будет принимать идентификатор книги и новые данные о книге, и обновлять информацию о книге в базе данных. Считайте текущее содержимое файла базы данных, преобразуйте его в надлежащий формат JSON, найдите соответствующую книгу по идентификатору и обновите ее данные. Затем запишите обновленные данные обратно в файл базы данных.
7. Создайте функцию **deleteBook()**, которая будет принимать идентификатор книги и удалять книгу из базы данных. Считайте текущее содержимое файла базы данных, преобразуйте его в надлежащий формат JSON, найдите соответствующую книгу по идентификатору и удалите ее из массива объектов. Затем запишите обновленные данные обратно в файл базы данных.
8. Создайте функцию **findBook()**, которая будет принимать параметр для поиска книги и производить поиск книги по базе данных. Считайте текущее содержимое файла базы данных, преобразуйте его в надлежащий формат JSON, найдите соответствующую книгу по **id**, **title**, **year** или **description** и выведите результат поиска в консоли. Если ни одной книги не найдено, то вывести соответствующее сообщение в консоли.
9. Создайте файл **index.js**, который будет служить точкой входа в программу.
10. В файле **index.js**, создайте интерфейс командной строки с помощью модуля **readline-sync**, который будет позволять пользователю взаимодействовать с программой.
11. В интерфейсе командной строки, предоставьте пользователю следующие команды:
    * "**[1]** **Add a new book**": Запрашивайте у пользователя данные о новой книге и вызывайте функцию **addBook()** для добавления книги в базу данных.
    * "**[2]** **View** **list of books**": Вызывайте функцию **getAllBooks()** и выводите список всех книг в консоль.
    * "**[3] Update existed book**": Запрашивайте у пользователя идентификатор книги и новые данные о книге, а затем вызывайте функцию **updateBook()** для обновления информации о книге в базе данных.
    * "**[4] Delete a book**": Запрашивайте у пользователя идентификатор книги и вызывайте функцию **deleteBook()** для удаления книги из базы данных.
    * "**[5] Find a book**": Запрашивайте у пользователя любые данные книги и вызывайте функцию **findBook()** для поиска книги в базе данных.
    * "**[0] Exit**": Завершайте работу программы.
12. Обработайте ошибки и предусмотрите соответствующие сообщения об ошибках в случае возникновения проблем при чтении или записи файлов базы данных.
13. Запустите программу и убедитесь, что вы можете успешно добавлять, получать, обновлять и удалять книги из базы данных.

**Примечание**:

Убедитесь, что база данных книг представлена в виде файла **books.json** в формате JSON.

**Список литературы:**1. Readline-sync - <https://www.npmjs.com/package/readline-sync>

2. FS - <https://nodejs.org/api/fs.html>