1 Practice 16.1. API Rest with Express

> This practice is the next of the series of practices, consisting of a Library application. Each practice will be based on the previous one. Before starting it, finish the previous one, make the corrections you consider, if needed, and start coding.
>
> Before start, create a new git branch named `books_review-u16` and switch to it:
>
> ```
> git branch books_review-u16
> git checkout books_review-u16
> ```
>
> Do all the practice in the `books_review-u16` branch. Once finished, do the submission via GitHub Classroom as detailed at the end of the practice.

In this practice we will create the API for our book's review app, using the database code of the previous practice.

1.1 Application overview

In this app we want to do a book's review API. In the database we will store a list of books and each book will contain a list of reviews.

In this practice we will create the database and operate with some data. It's recommended that you use the queries of the previous practice for each route.

1.2 Exercise 1. REST Web Services

Build the next web services:

Action	Method	Route	Notes
List of all books with their reviews	GET	/books/	
Single book by ID	GET	/books/{id}	
Single book by ISBN	GET	/books/isbn/{isbn}	
Search books by title	GET	/books/search/{search_string}	

Action	Method	Route	Notes
List of all the reviews of a book, by the book's ID	GET	/reviews/book/{id_book}	Show all the reviews data but not the book data
Insert a new book with their reviews	POST	/books/	
Update a book by its ID	PUT	/books/{id}	
Delete a book by its ID	DELETE	/books/{id}	
Insert a new review on a book identified by the book's ID	POST	/reviews/{book_id}	

All the books must be shown with all the data and their reviews.

Start the server in the **port 8080**.

1.3 How to submit to GitHub Classroom

Once you finish the task, make a commit with the comment "PRACTICE 16.1 SUBMISSION COMMIT" and push it to GitHub. Make a **pull-request** so that i could know your code is submitted.

1. This task must be submitted to the same repository of the previous one. Remember to make a new branch before starting the practice.
2. Once you finish the task, make a commit with the comment "PRACTICE 16.1 SUBMISSION COMMIT", merge the branch into the main branch, and push it to GitHub. For example:

```
git commit -m "PRACTICE 16.1 SUBMISSION COMMIT"
git checkout main
git merge books_review-u16
git push
```

3. It's recommended to tag your commit with the tag "Practice_16.1".

4. Before that, you can do the commits and push you want. If you change your code after your submission commit, make another commit and push with the same text in the message adding the corrections you've done.

If you have any doubt in your task, you can push your code and ask me by email what's your problem. It will make it easier for both the solutions of code issues.