

#BOOK FINDER API

Get information about Books and Authors via this BookFinder API.

#ABOUT THIS PROJECT

This REST API project Book Finder is providing detailed information about Books and Authors.

A user can add new books and authors, search for detailed information about a book, for example title, genre or info.

It is also possible to update and delete books and authors.

#BOOK FINDER

You can access API through URL <http://localhost:3000/api/>

In order to get a faster response, you can use the described REST endpoints below.

#API ENDPOINTS -

#ALL Books and Authors

<http://localhost:3000/api/books/>

<http://localhost:3000/api/authors/>

#Add a book

<http://localhost:3000/api/books/{id}>

This POST request needs the request body to be added with the new book info according to the bookObject model.

#Add an author

<http://localhost:3000/api/authors/{id}>

This POST request needs the request body to be added with the new author info according to the authorObject model.

Filtering Endpoints

#Search by book id to find a specific book

<http://localhost:3000/api/books/{id}>

#Search by author id to find a specific author

<http://localhost:3000/api/authors/{id}>

#Search a book by Title

<http://localhost:3000/api/books/{id}?fields=title>

#Search a book by Year

<http://localhost:3000/api/books/{id}?fields=year>

#Search a book by Info

<http://localhost:3000/api/books/{id}?fields=info>

#Update a book

<http://localhost:3000/api/books/{id}>

This PUT request needs the request body to be updated with the new info for the book according to the bookObject model.

#Update an author

<http://localhost:3000/api/authors/{id}>

This PUT request needs the request body to be updated with the new info for the author according to the authorObject model.

#Delete a book

<http://localhost:3000/api/books/{id}>

This DEL request needs to be done on a specific book id.

#Delete an author

<http://localhost:3000/api/authors/{id}>

This DEL request needs to be done on specific book id.

#Pagination

<http://localhost:3000/api/books/all?page=1&limit=10>

Pagination is available for books.

Page and limit are specified in the GET request parameters. In this case 10 books will be presented on the first page.