

BookTracker2000 Endpoints

Av Malte Nilsson

AuthorsController (api/authors)

GET

Returns a list of all authors in the database as AuthorViews.

GET ({ id })

Returns the specified author as an AuthorView.

PUT ({ id })

Updates the specified author's first name and/or last name.

POST

Adds a new author to the database, with the required first name and last name.

DELETE ({ id })

Deletes the specified author from the database.

BooksController (api/books)

GET

Returns all the books in the database as BookViews.

GET ({ id })

Returns the specified book as a BookView.

PUT ({ id })

Updates the specified book's title, total pages, and/or authors. Removes all the previous authors. The author must already exist.

PUT ({ bookId }/authors/add)

Adds a new author to the author list of the specified book. The author must already exist.

POST

Adds a new book to the database, with the required author, title, and total pages.

DELETE ({ id })

Deletes the specified book from the database.

DELETE ({ bookId }/authors/remove/{ authorId })

Deletes the specified author from the book, UNLESS it's the last author.

ReadingsController (api/readings)

GET

Returns all the readings in the database as ReadingViews.

GET ({ id })

Returns the specified reading as a ReadingView.

GET (current)

Gets all the readings of the current user as ReadingViews.

PUT ({ id })

Updates the specified reading's read pages.

POST

Adds a new reading, with the required user-book and pages read.

DELETE ({ id })

Deletes the specified reading from the database.

UserController (api/users)**GET**

Returns all the users in the database as UserViews.

GET ({ id })

Returns the specified user as a UserView.

GET (current)

Returns the current user as a UserView.

PUT ({ id })

Updates the specified user with username, password hash, and/or email.

POST (authenticate)

Authenticates a username and password and returns a jwt.

POST (register)

Adds a user to the database with a username, password hash and email.

DELETE ({ id })

Deletes the specified user from the database.

DELETE (current)

Deletes the current user from the database.

DELETE ({ userId }/books/{ bookId })

Deletes a specified book from the specified user's booklist.

DELETE (current/books/{ bookId })

Delete the specified book from the current users booklist.