CMPS 350- Web Development Fundamentals

**Assignment 4 Web APIs**

|  |  |
| --- | --- |
| **Student Name** | Syed Ahmad Faraz Subzwari |
| **Student Id** | 202105006 |
| **Email** | SS2105006@qu.edu.qa |

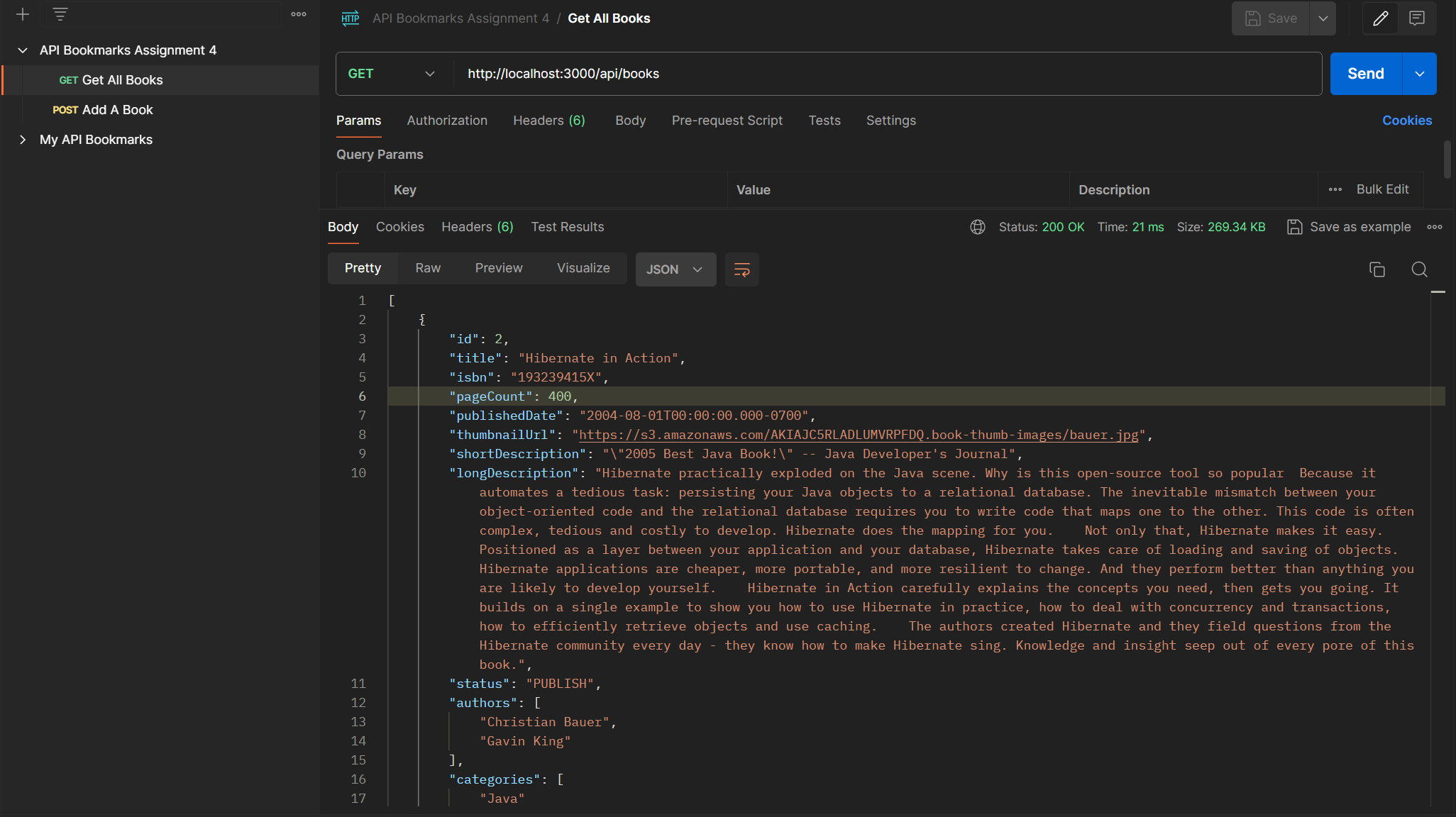
**Grading Rubric - - In the *Functionality* column please specify either: *Working (completed x%)*, *Not Working (completed x%)* or *Not done*.**

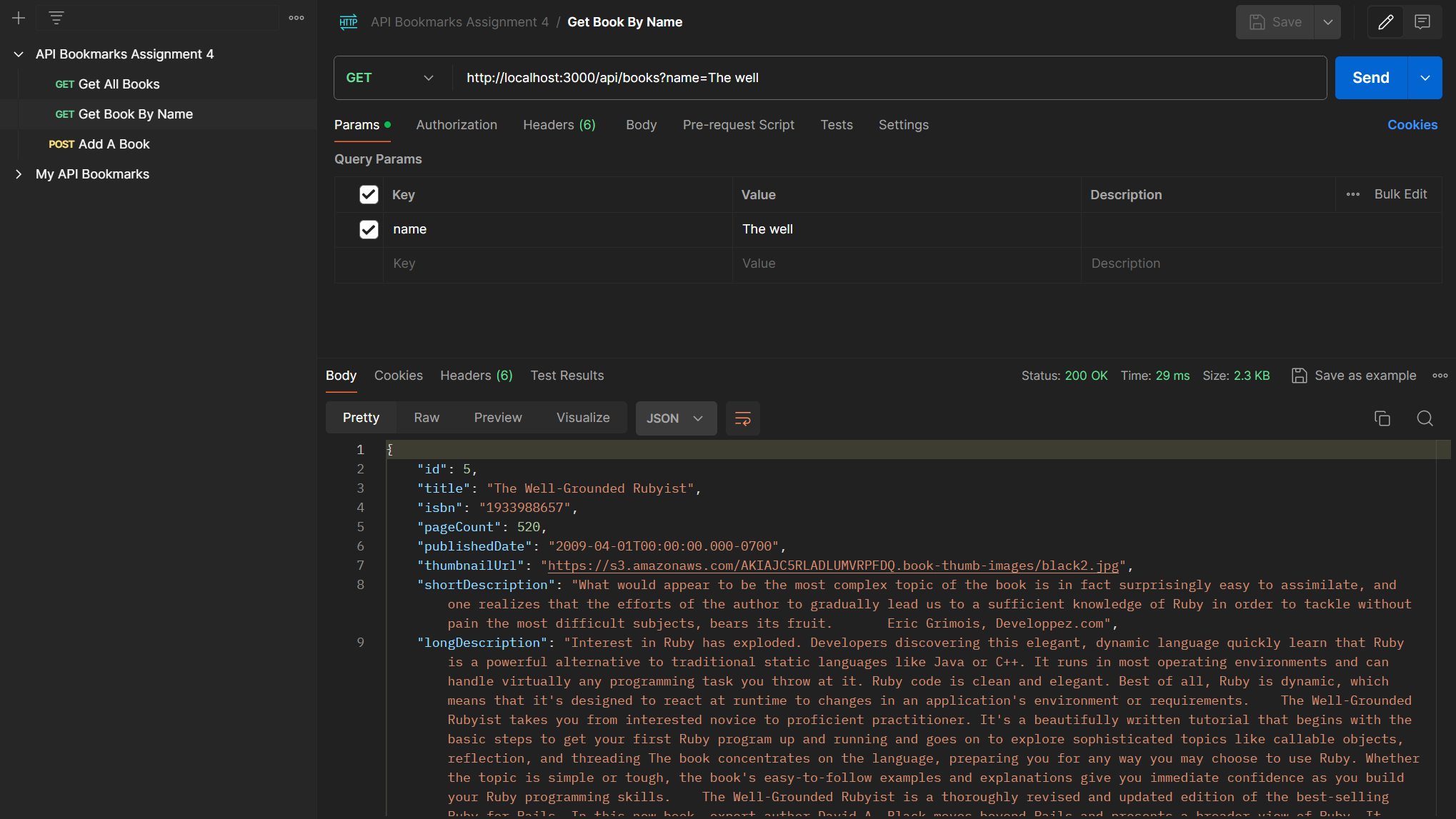
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Criteria | % | Functionality\* | Quality of the implementation | Score |
| Part B - Make the Book Store App functionality accessible via Web API. | 85 | 85 Working(completed 100%) |  |  |
| 1. Get api/books 2. Get api/books?name= 3. Get /api/books?pageCount= 4. Get /api/books?author= 5. Get /api/books?category= 6. Get /api/books/summary 7. Post /api/books/ 8. Put /api/books/:isbn 9. Delete /api/books/:isbn |  |  |  |  |
| Testing using POSTMAN , Chai-HTTP and the UI | 15 | 15 Working(completed 100%) |  |  |
| Total |  |  |  | 100 |
| Copying and/or plagiarism or not being able to explain or answer questions about the implementation | -100 |  |  |  |

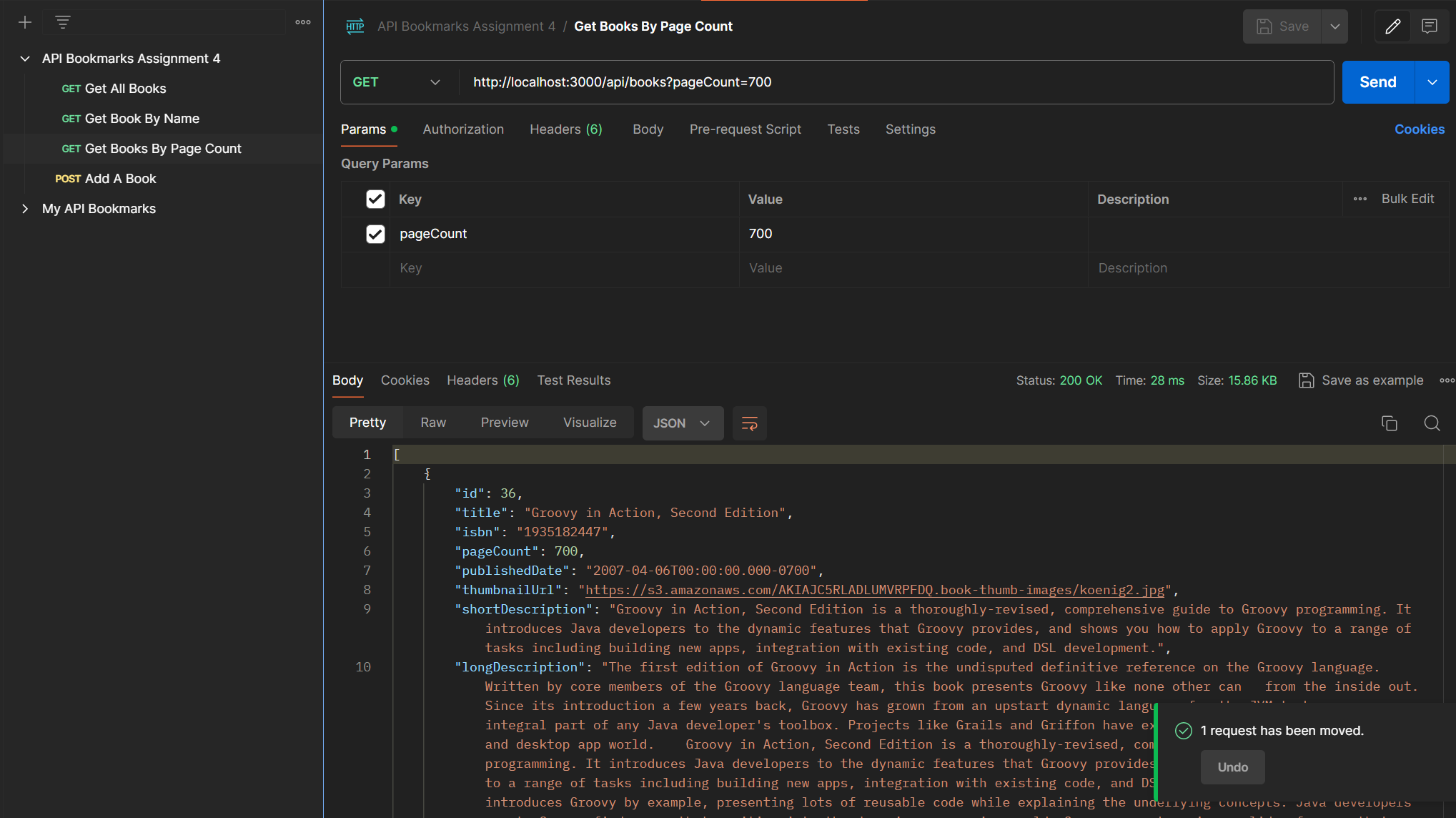
**\* Possible grading for functionality**: ***Complete and*** ***Working*** (get 70% of the assigned grade), ***Complete and*** ***Not*** ***working*** (lose 40% of assigned grade) and ***Not done*** get 0. The remaining grade is assigned to the quality of the implementation. In case your implementation is not working then 40% of the grade will be lost and the remaining 60% will be determined based on of the code quality and how close your solution to the working implementation. Quality includes meaningful naming of identifiers, no redundant code, simple and efficient design, clean code without unnecessary files/code, use of comments where necessary, proper white space and indentation. **Marks will be reduced** forcode duplication, poor/inefficient coding practices, poor naming of identifiers and unnecessary complex/poor user interface design.

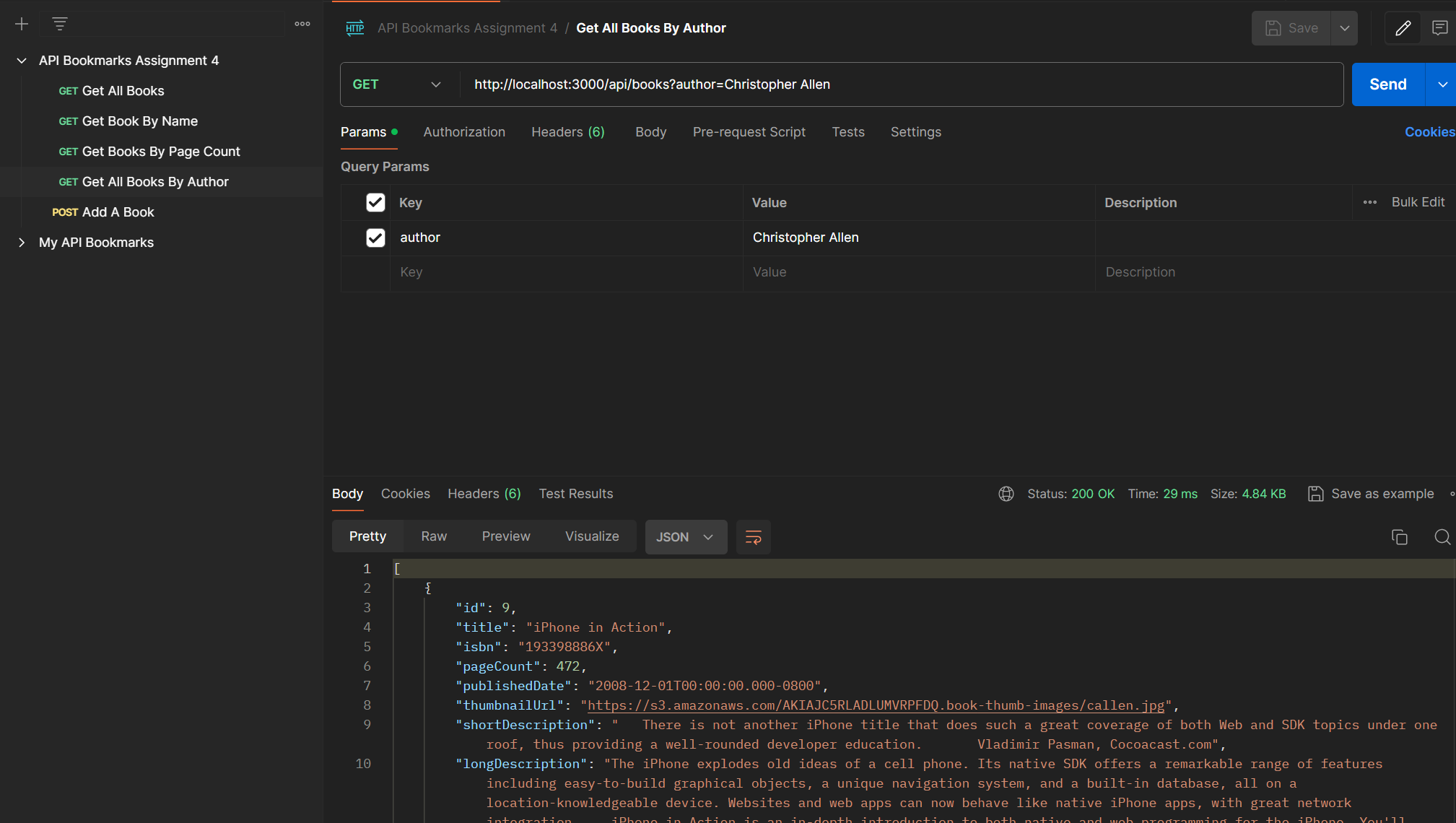
## Testing evidence

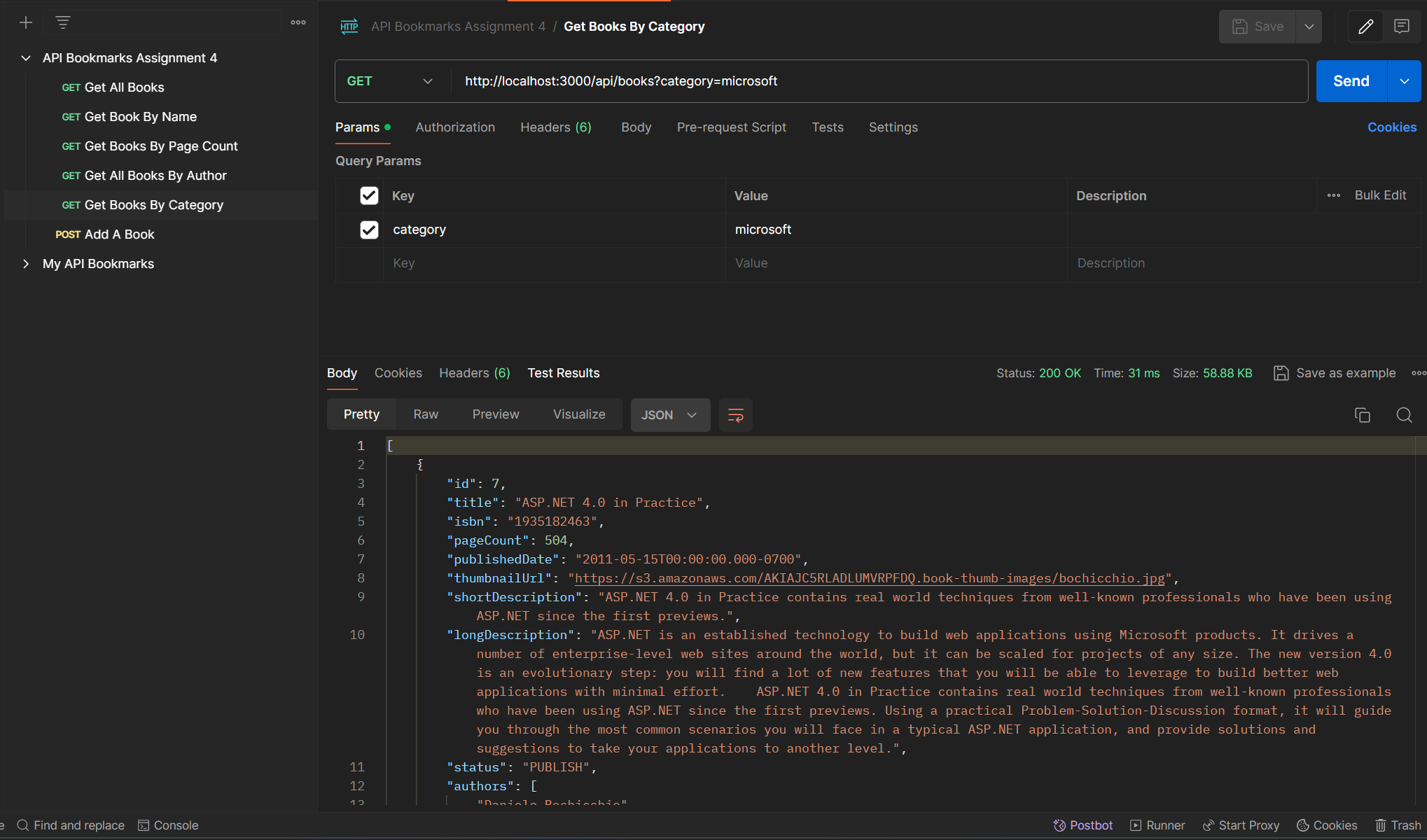
## Postman:

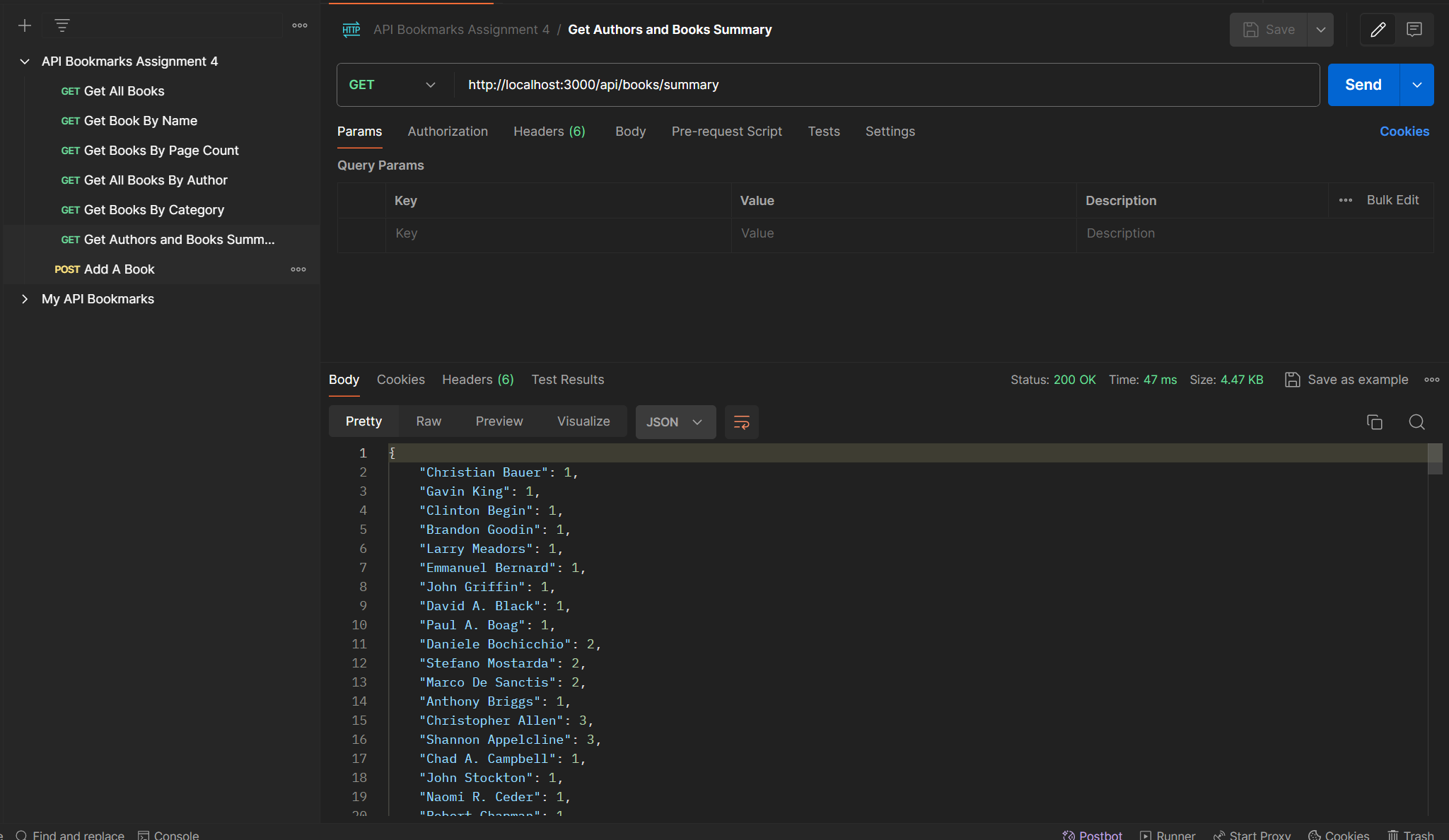
**Getting all the Books:** As we can see, all the books are returned.

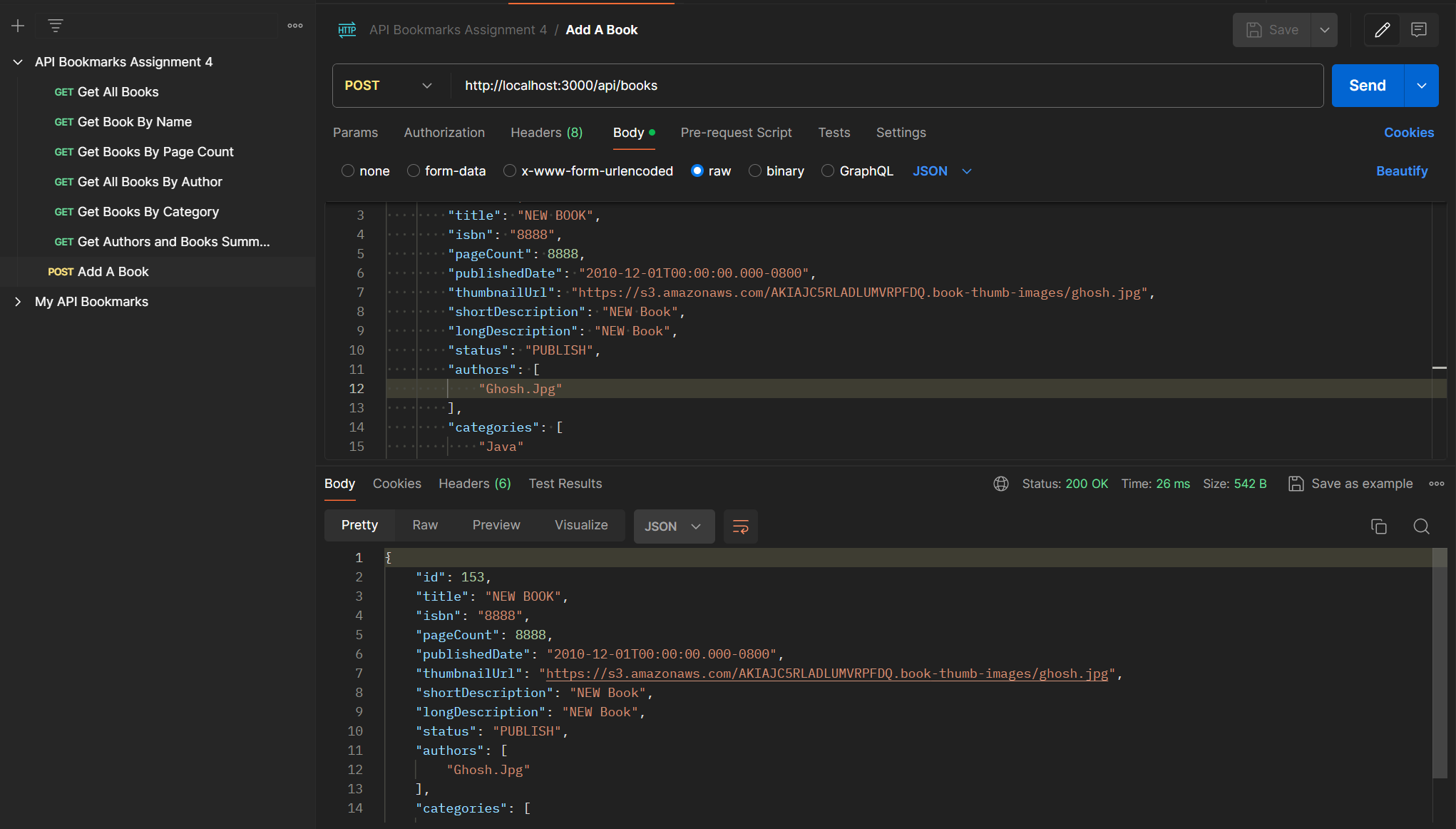
**Getting a Book by Name:** Only the book that contained “The Well” is returned

**Get Books by Page Count:** All the books that have a page count equal or higher than 700 are returned.

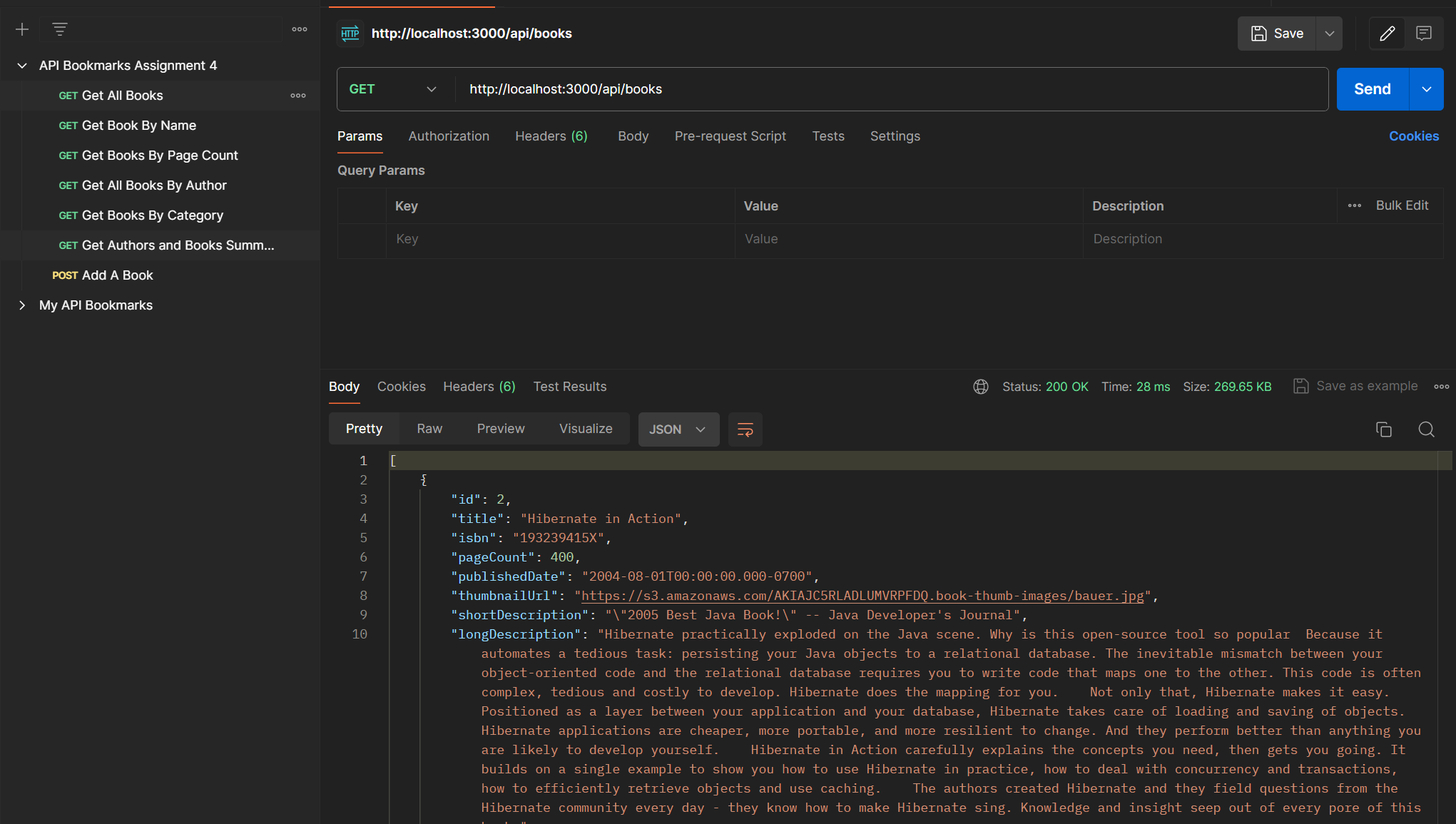
**Get Books by Author:** All books that are by this author (Christopher Allen) are returned. 

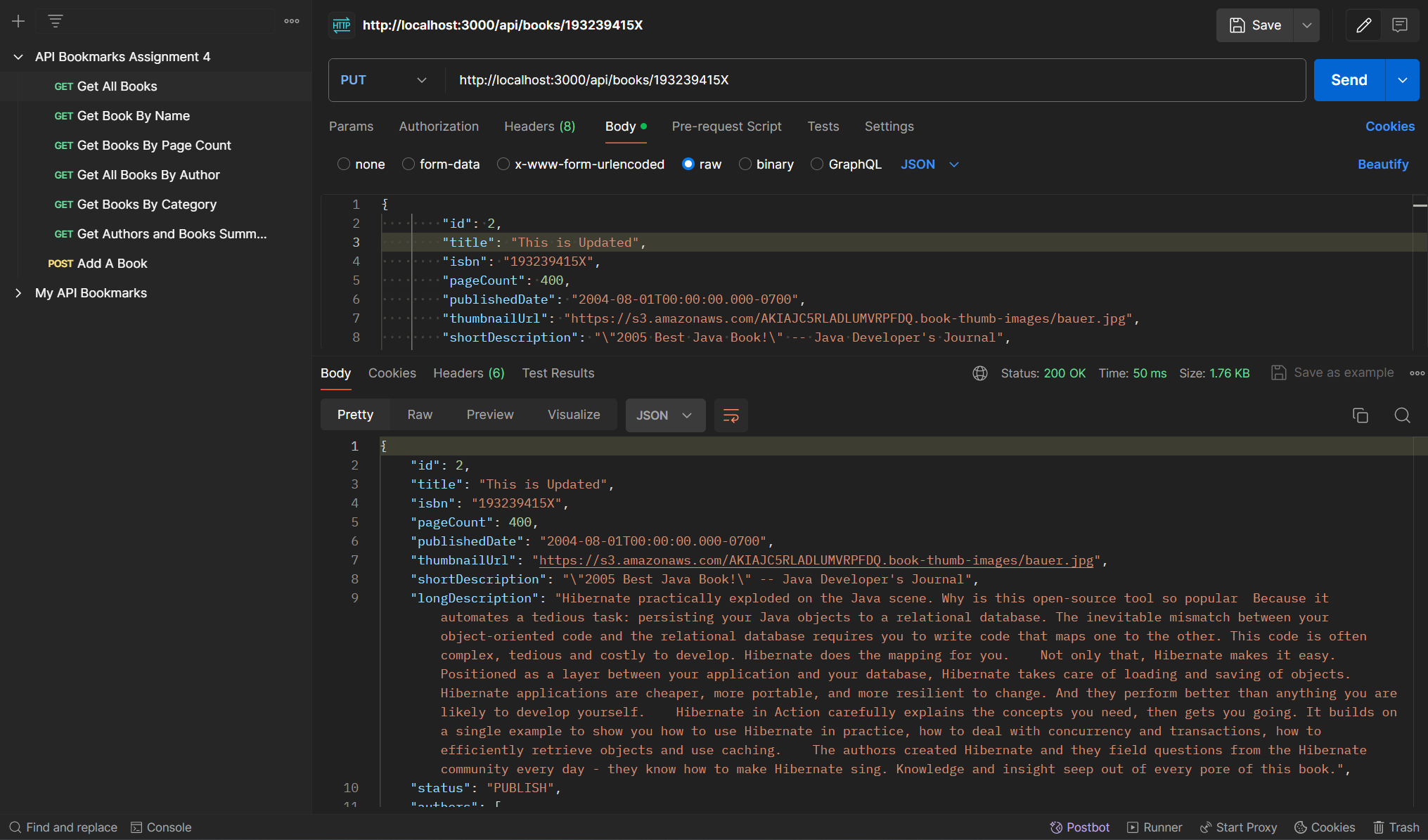
**Get Books by Category:** All books from a specific category are returned. 

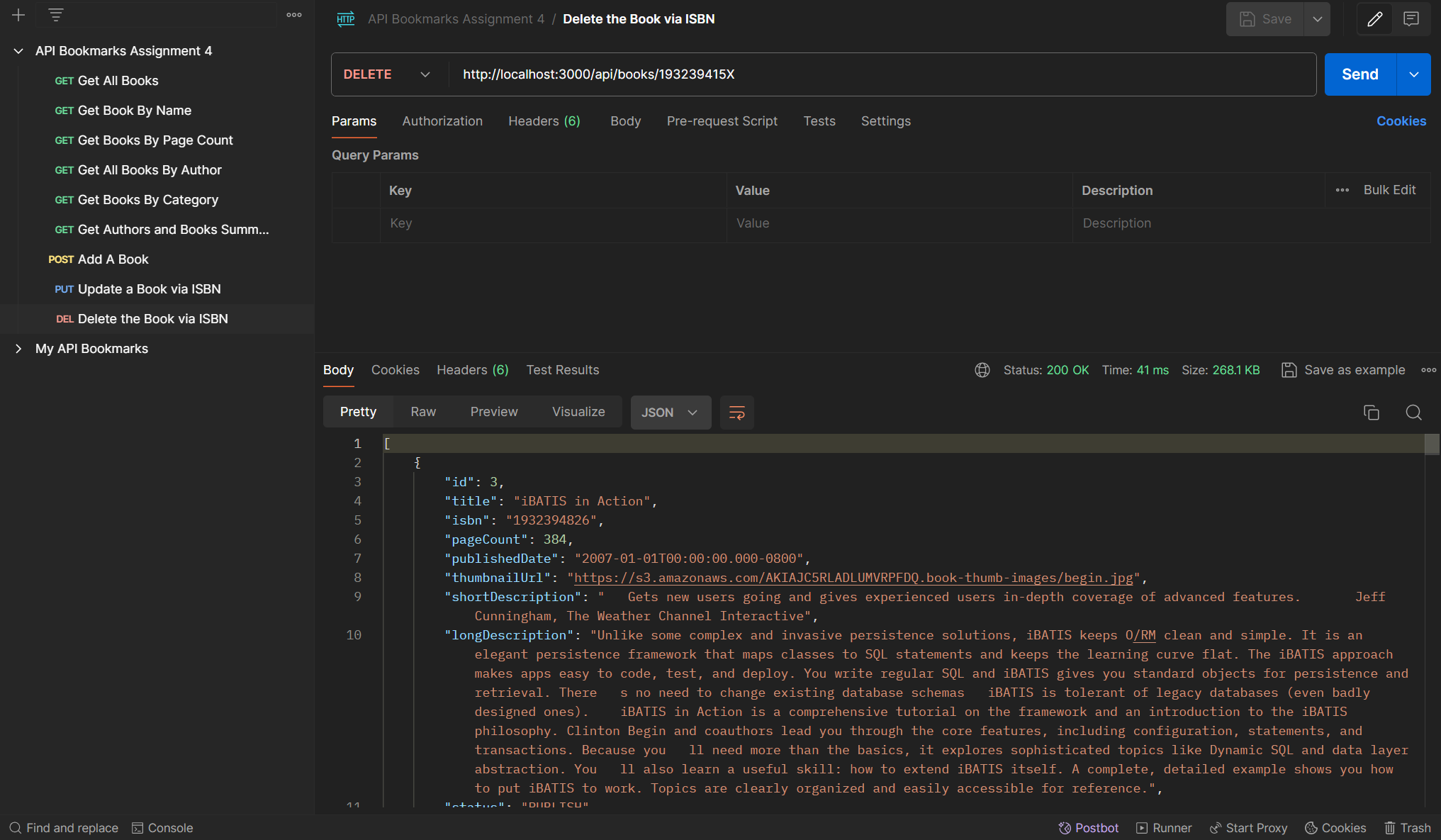
**Get Summary:** The Authors and their Books are returned as Summary. 

**Post a book:** Adding a book to the List. Added book is returned and the following book can also be seen by the GET method of All books.

**Put a book:** Updating a book’s details. Uses ISBN as a unique identifier. This is the book I will be changing. Let’s say that I want to change the title of the book to “This is Updated”.

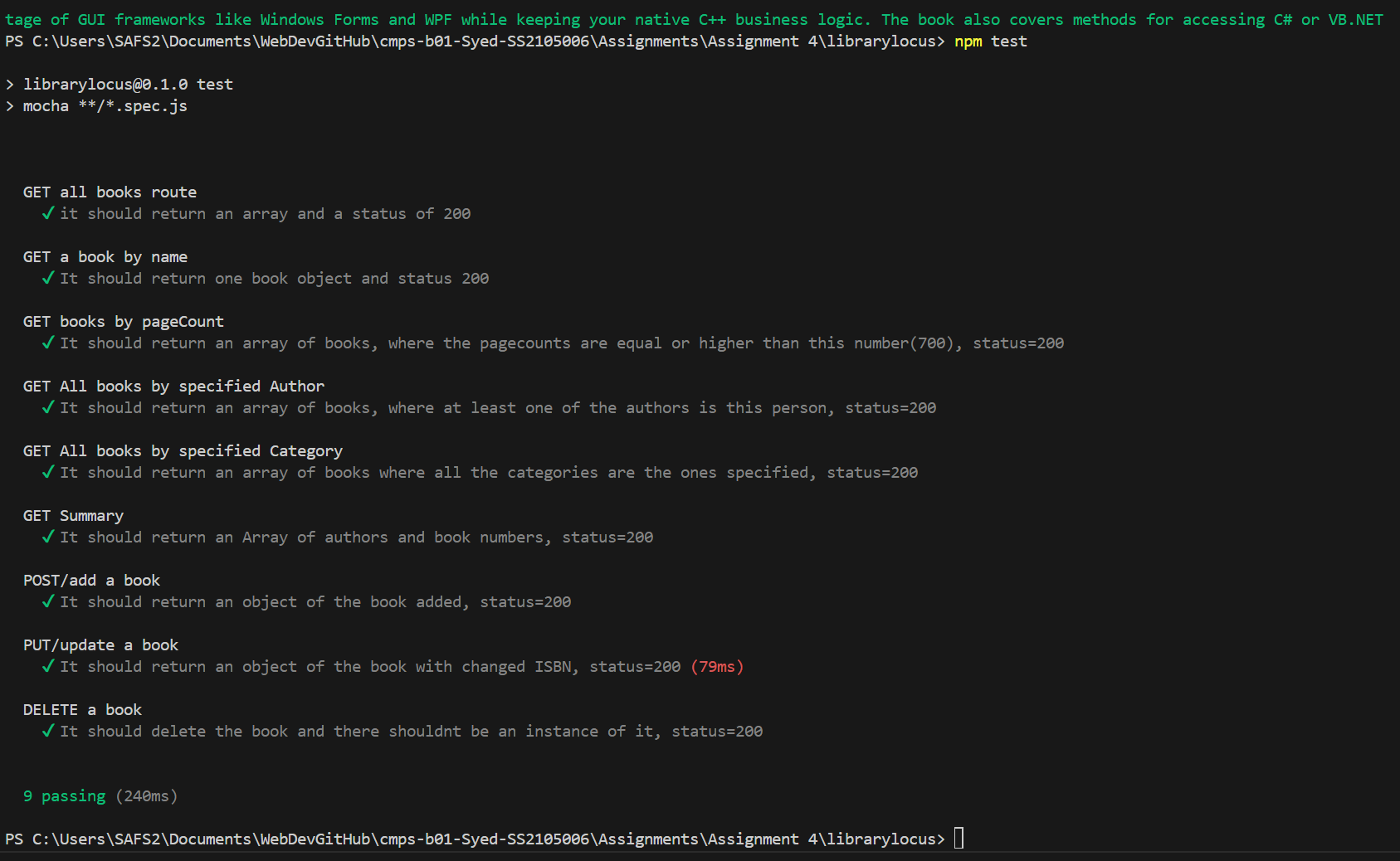


We can see that the title has been changed to “This is Updated”. 

**Delete a Book:** We will be deleting the book via the ISBN. I will delete the Book with ID: 2 that we saw in the previous image. 

As we can see, the rest of the list after deletion is shown. Hence, the book has indeed been deleted.

## Chai-HTTP:



All Tests passed and completed.