**Exercise 8: Online Bookstore - Implementing CRUD Operations**

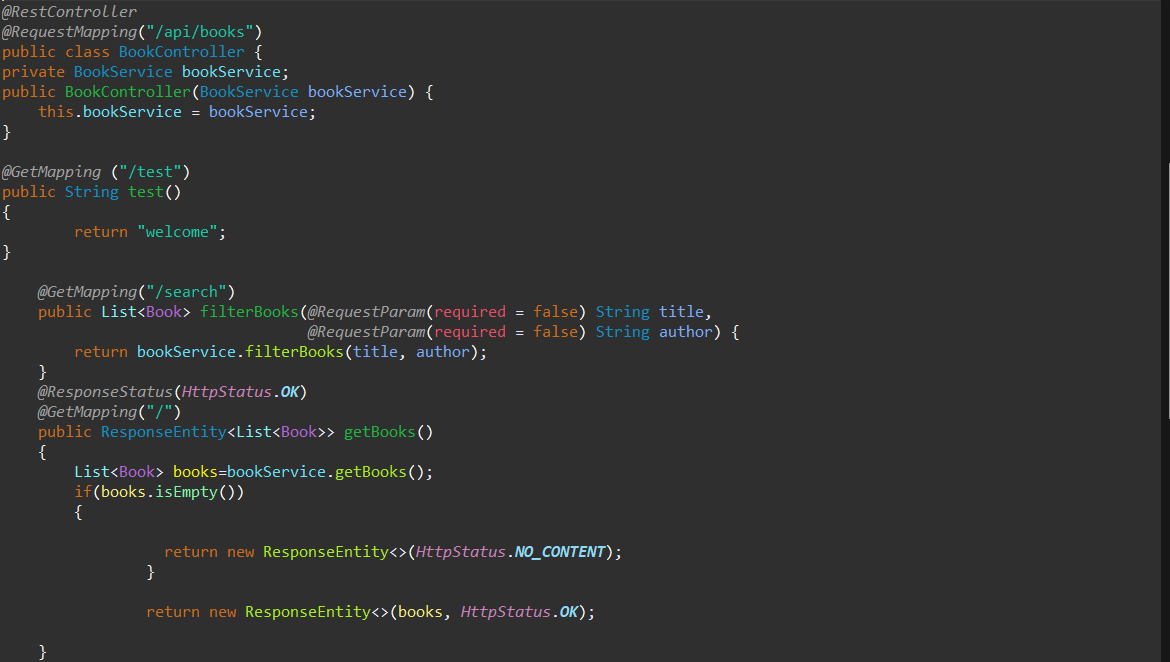
Business Scenario:

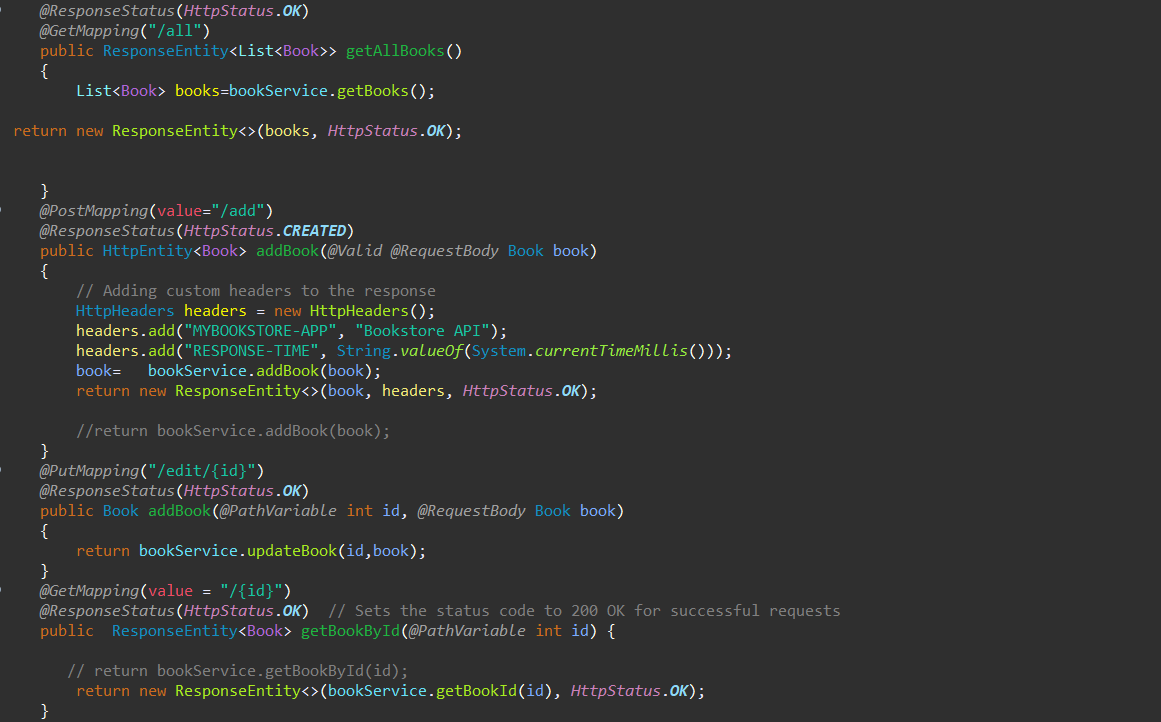
Implement Create, Read, Update, and Delete operations for the Book and Customer entities.

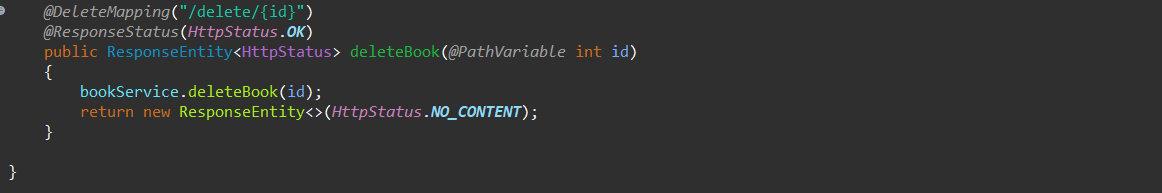
1. **CRUD Endpoints:**

I implemented endpoints for creating, reading, updating, and deleting books and customers in the BookController and CustomerController classes respectively

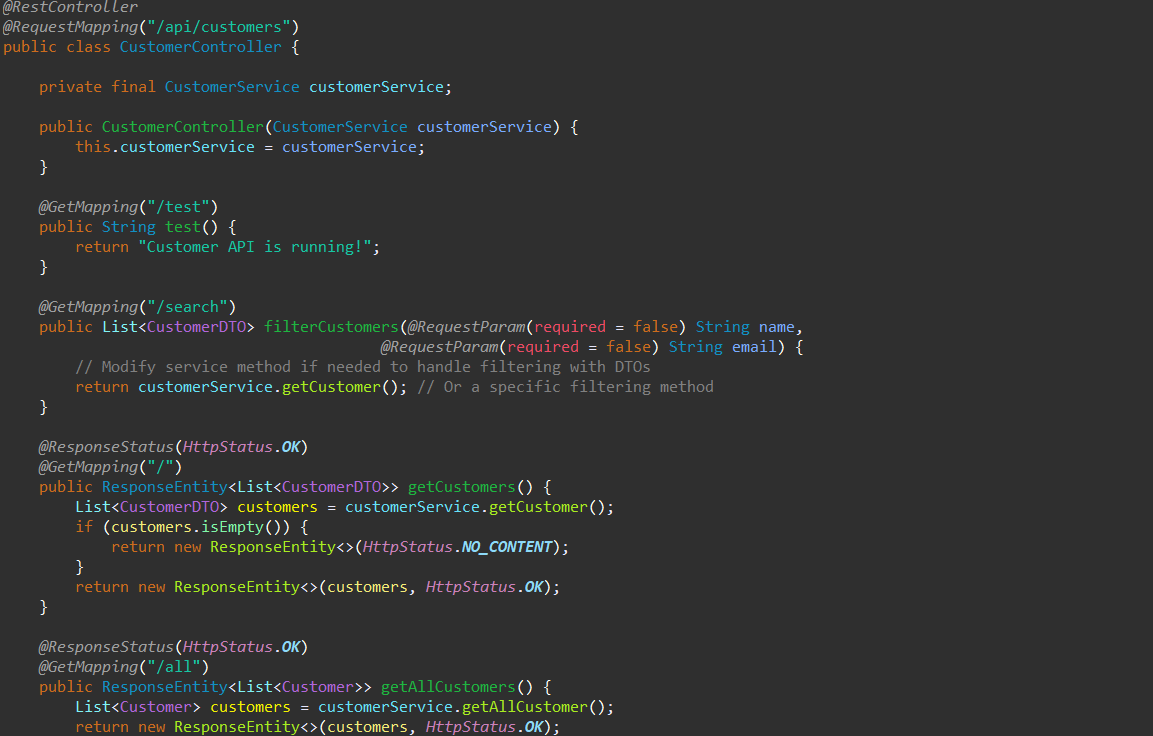
BookController.java:

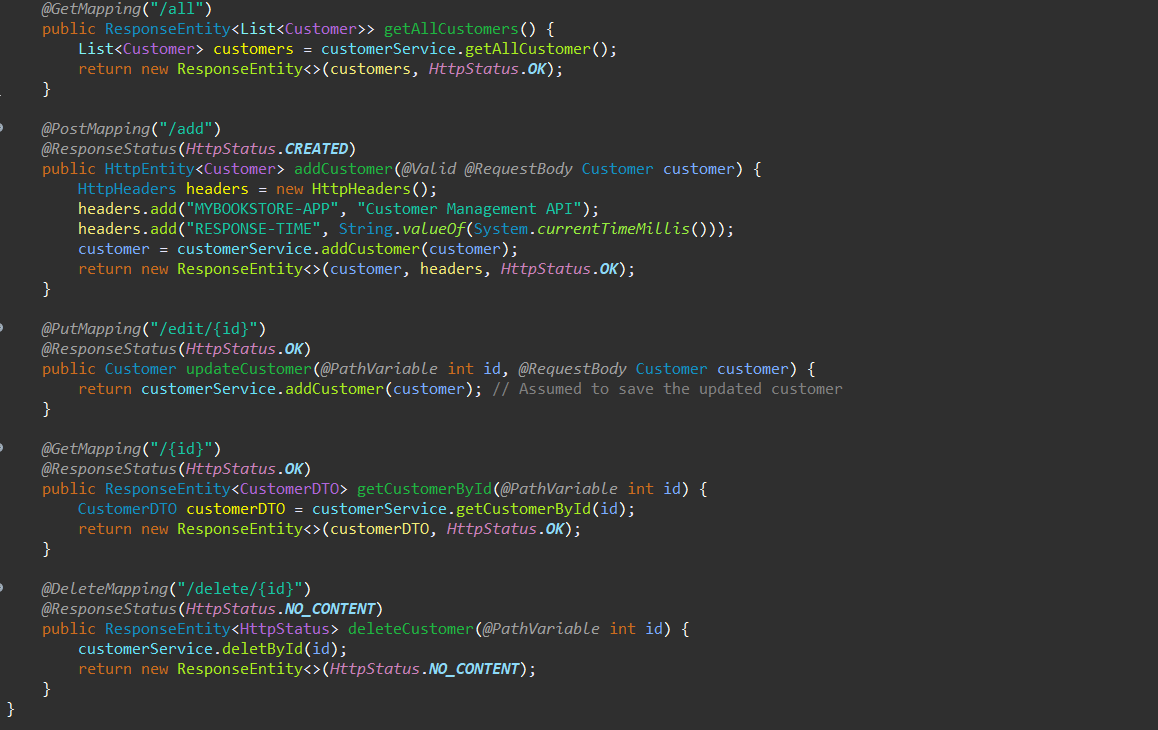






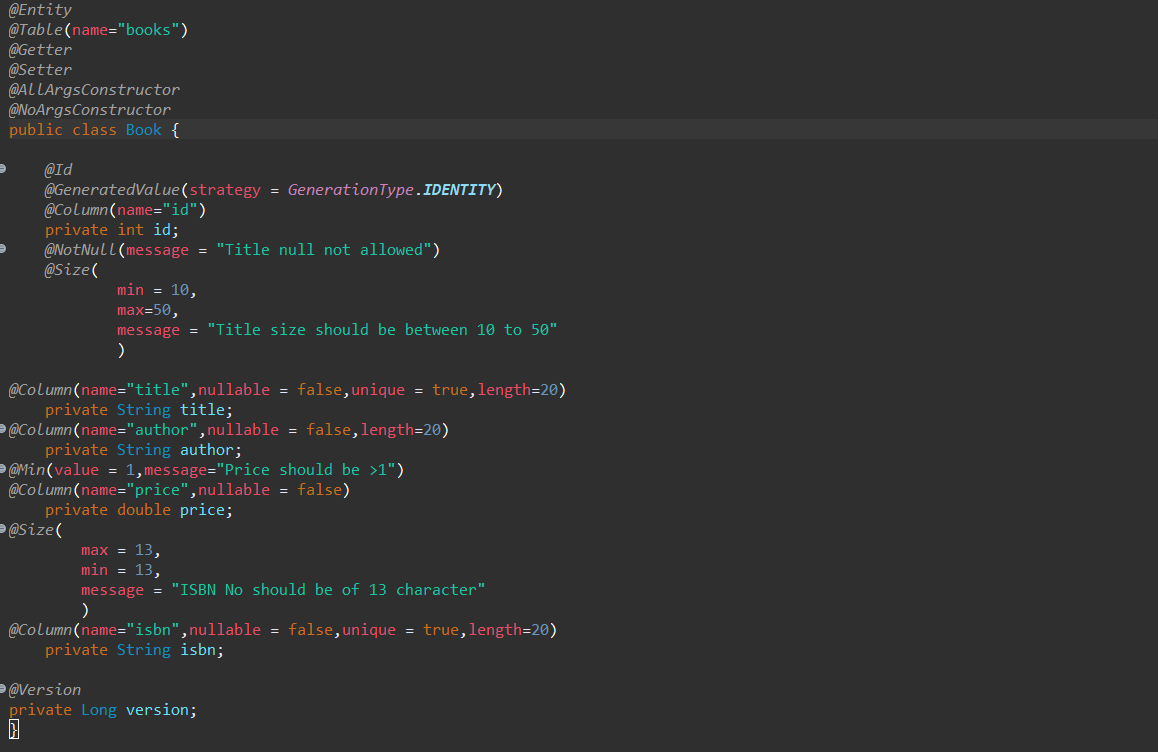
CustomerController.java

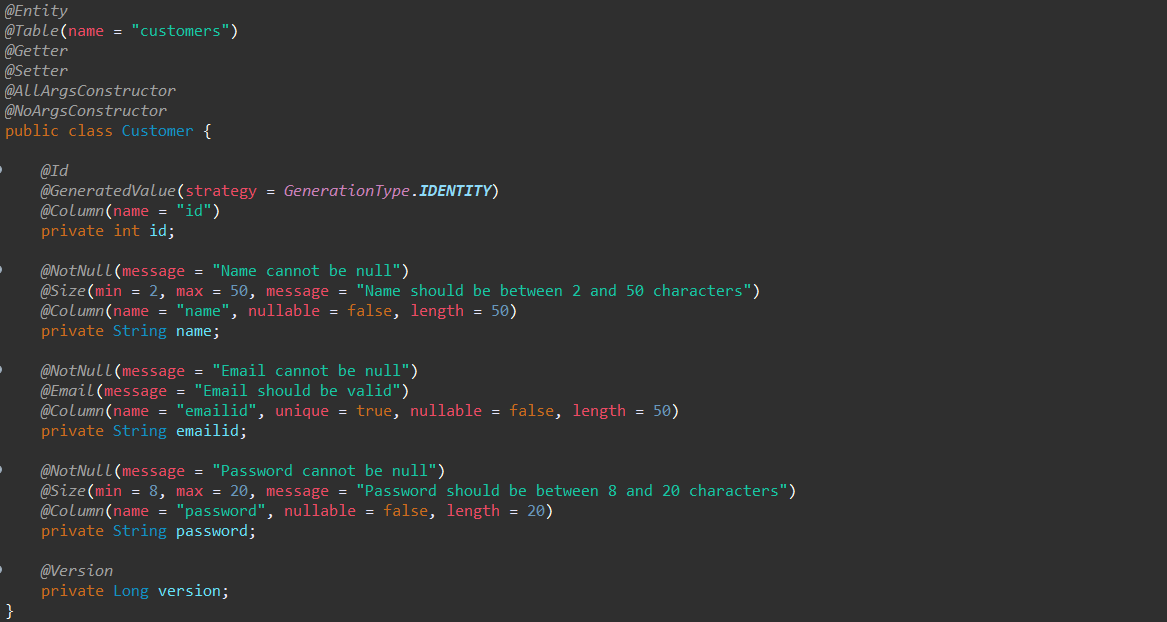




1. Validating input data:

* I used validation annotations like **@NotNull, @Size**, and **@Min** to validate input data for both Book class and Customer class



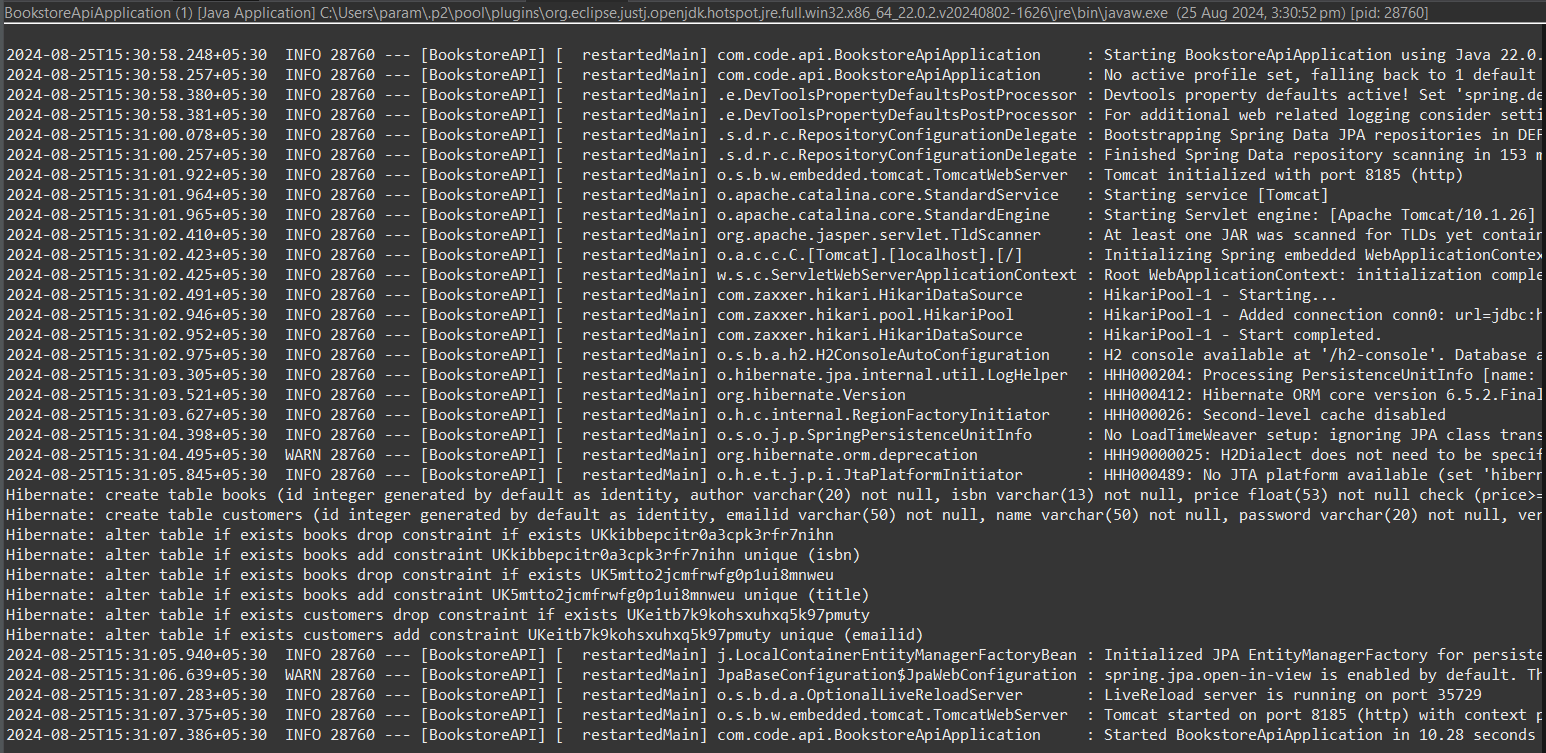


1. Optimistic Locking:

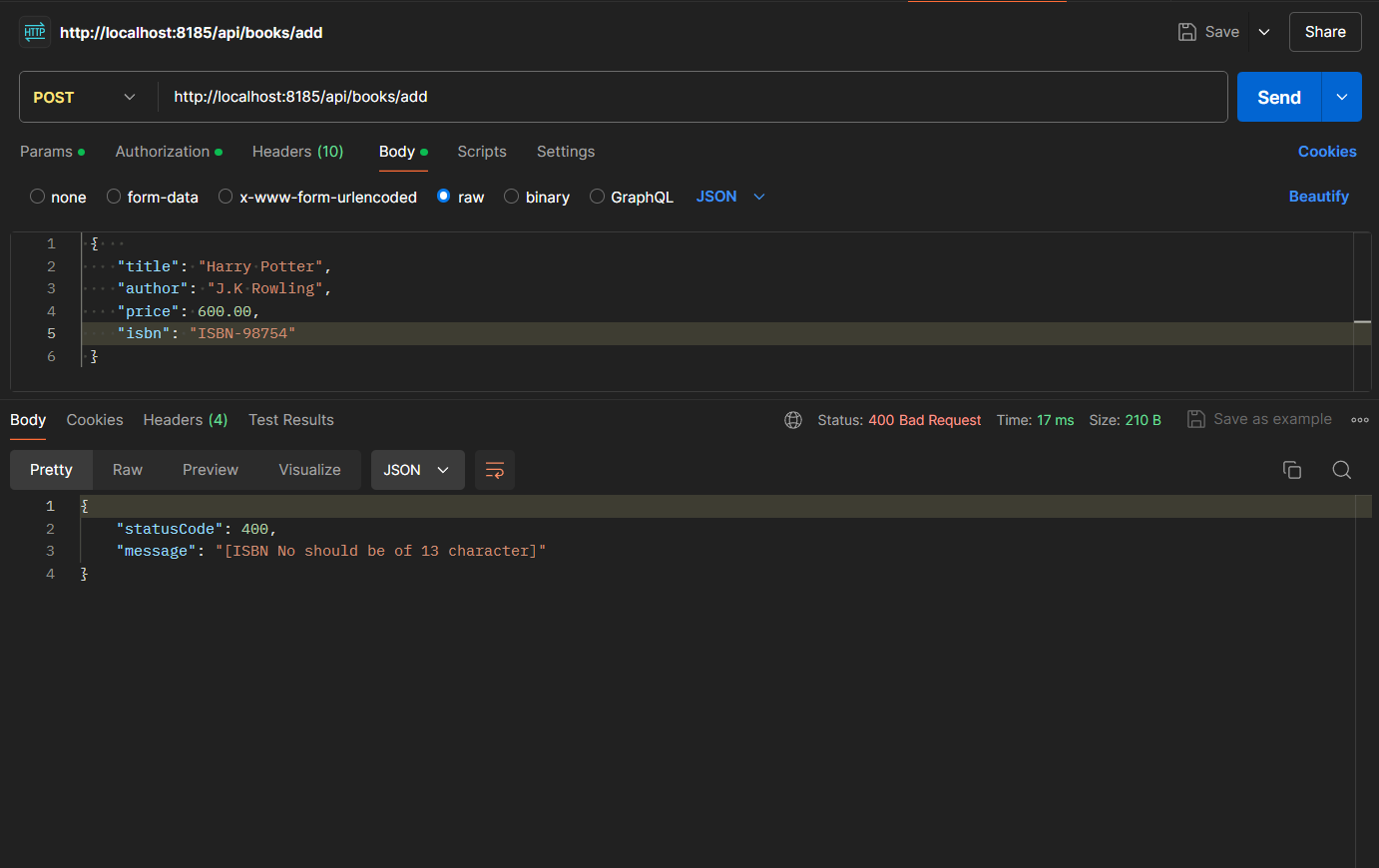
* I implemented optimistic locking for concurrent updates using JPA Versioning using @Version in both Book and Customer classes



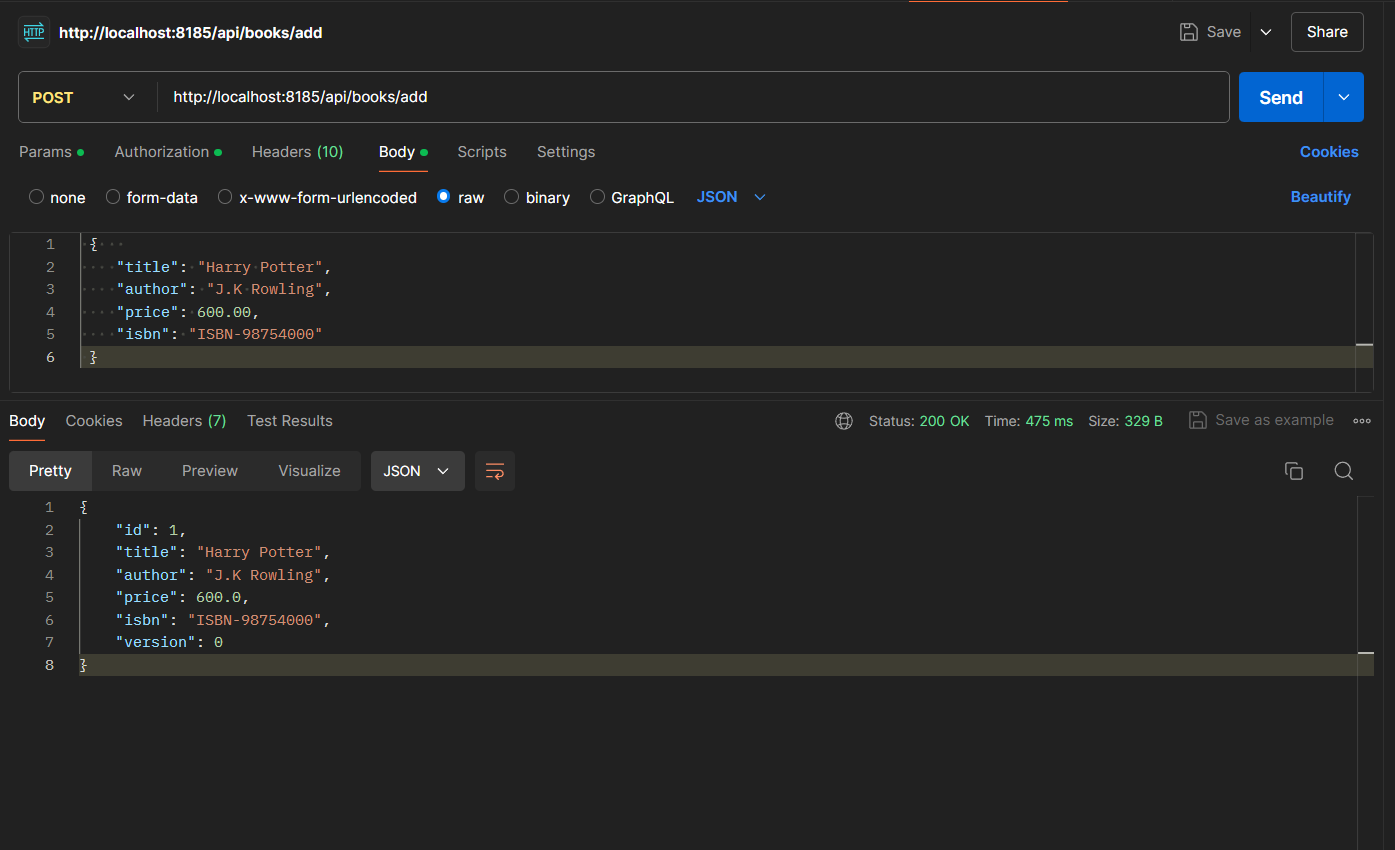
**Output:**



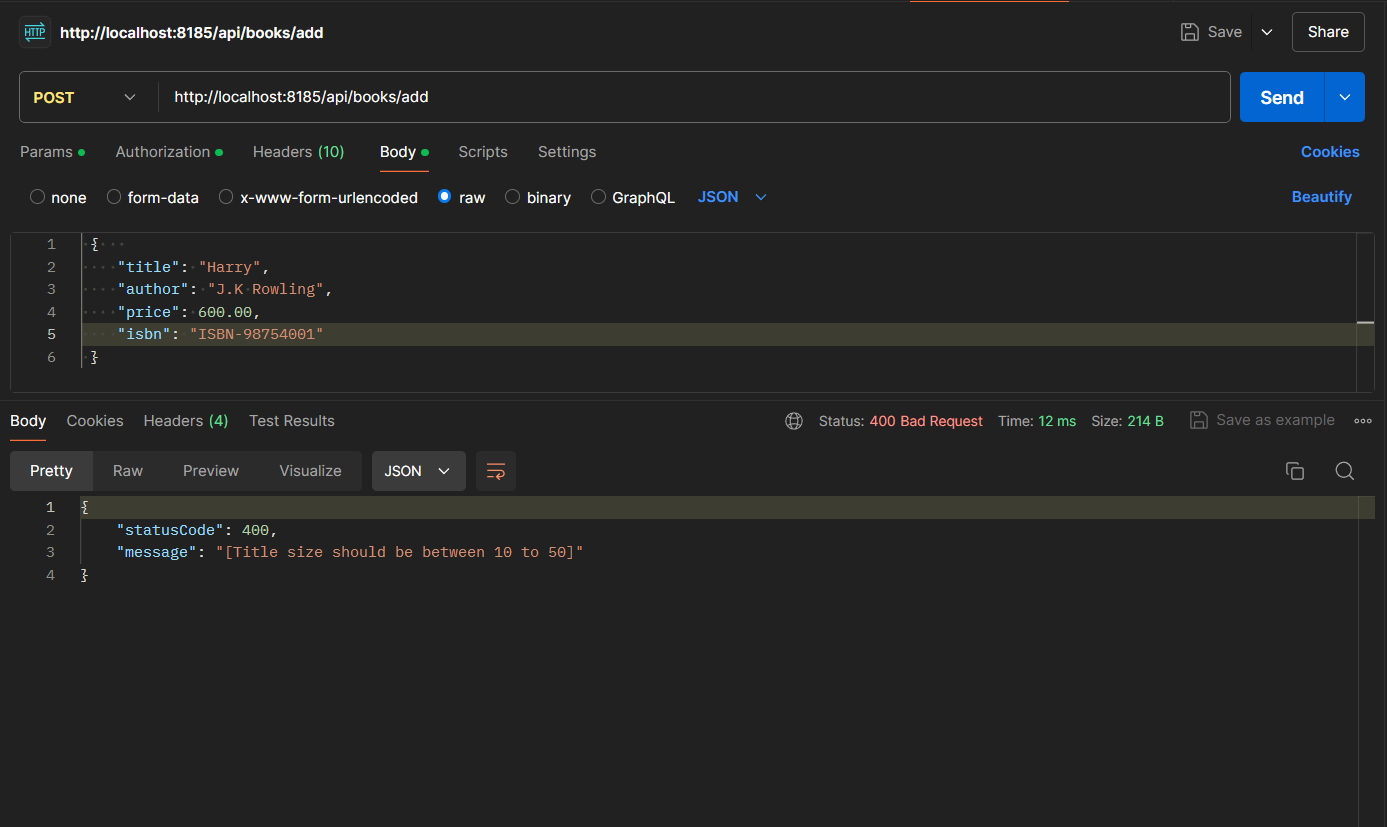
While adding book if the ISBN is below 13 characters:



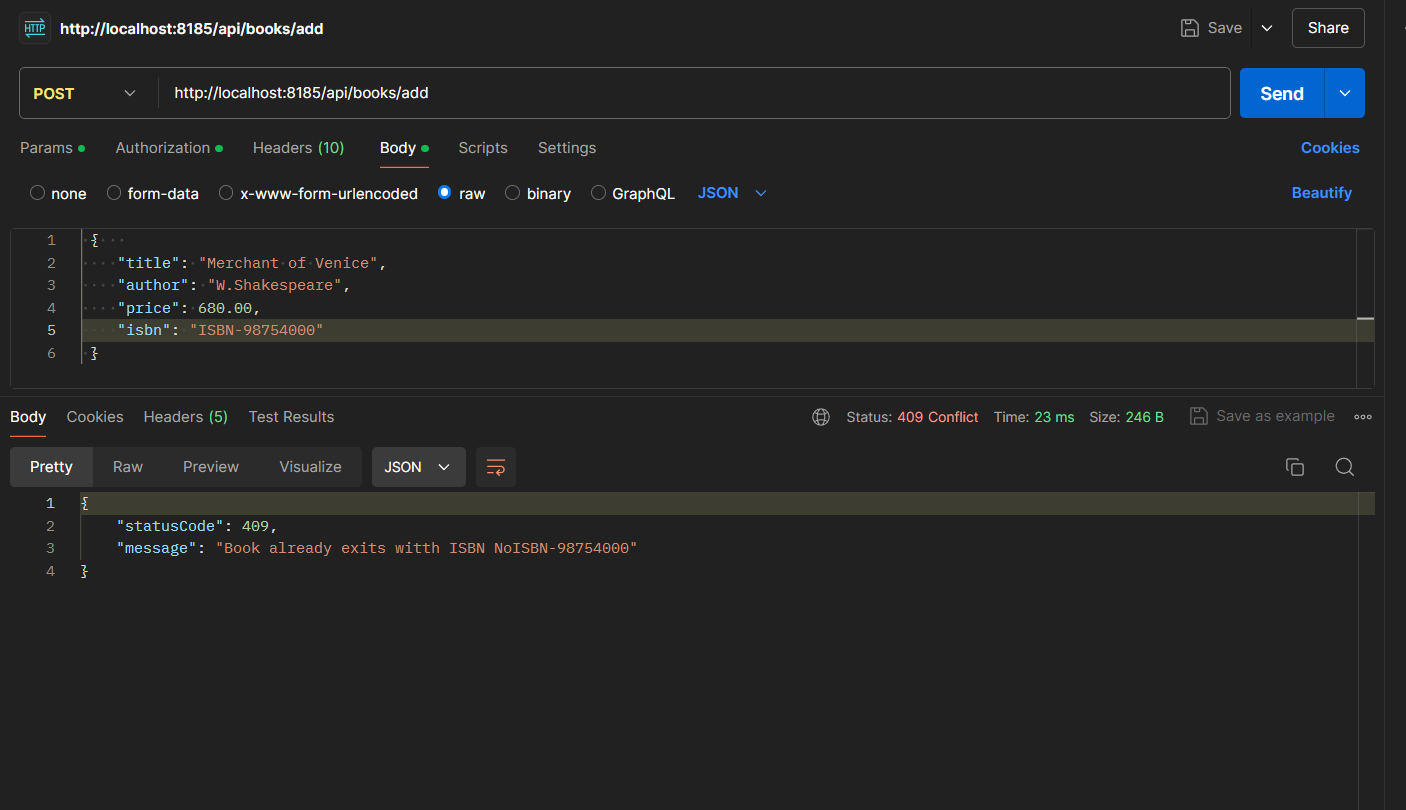
Add Books:



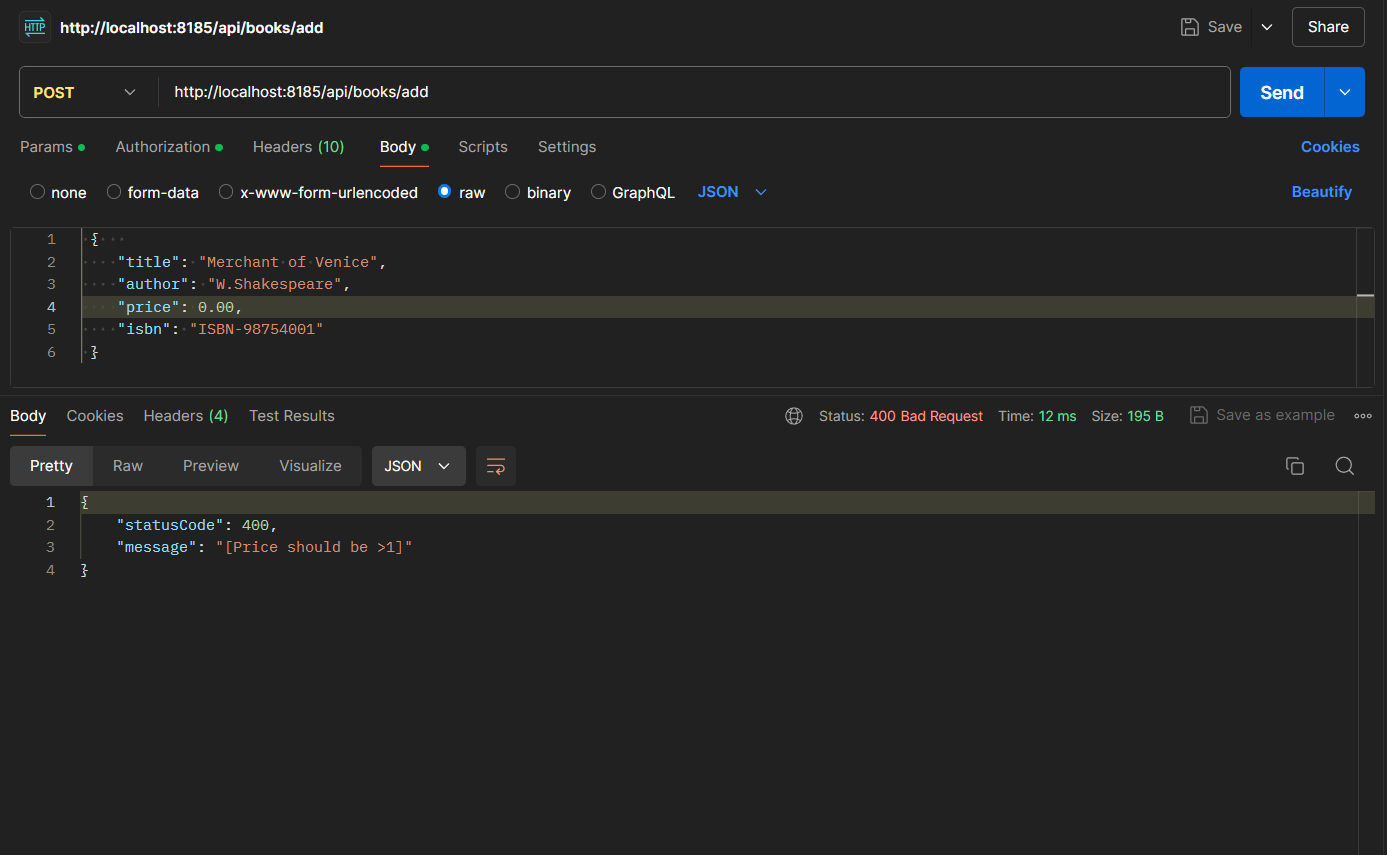
While adding book if the title is below 10 characters:



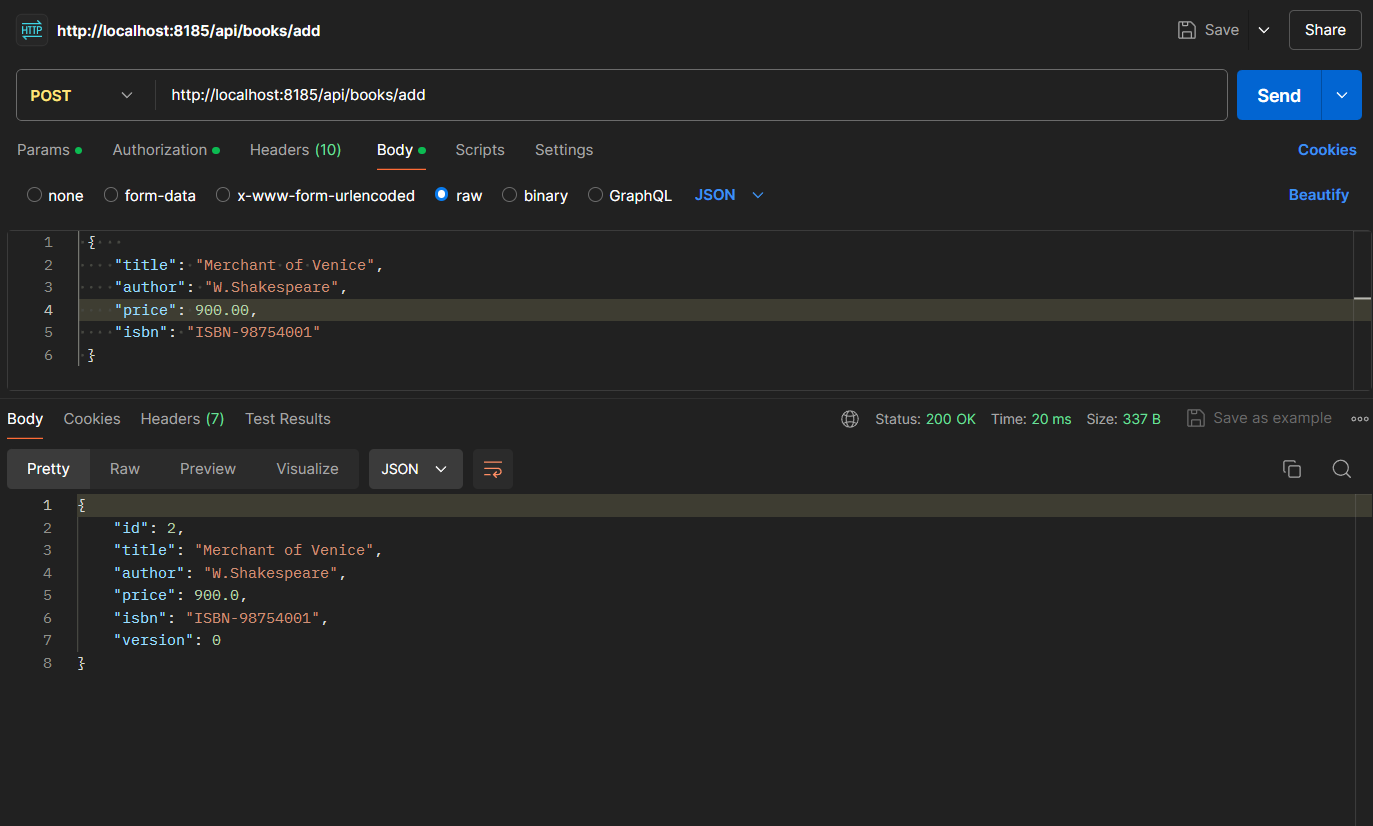
If book to be added is having same ISBN as of another book:



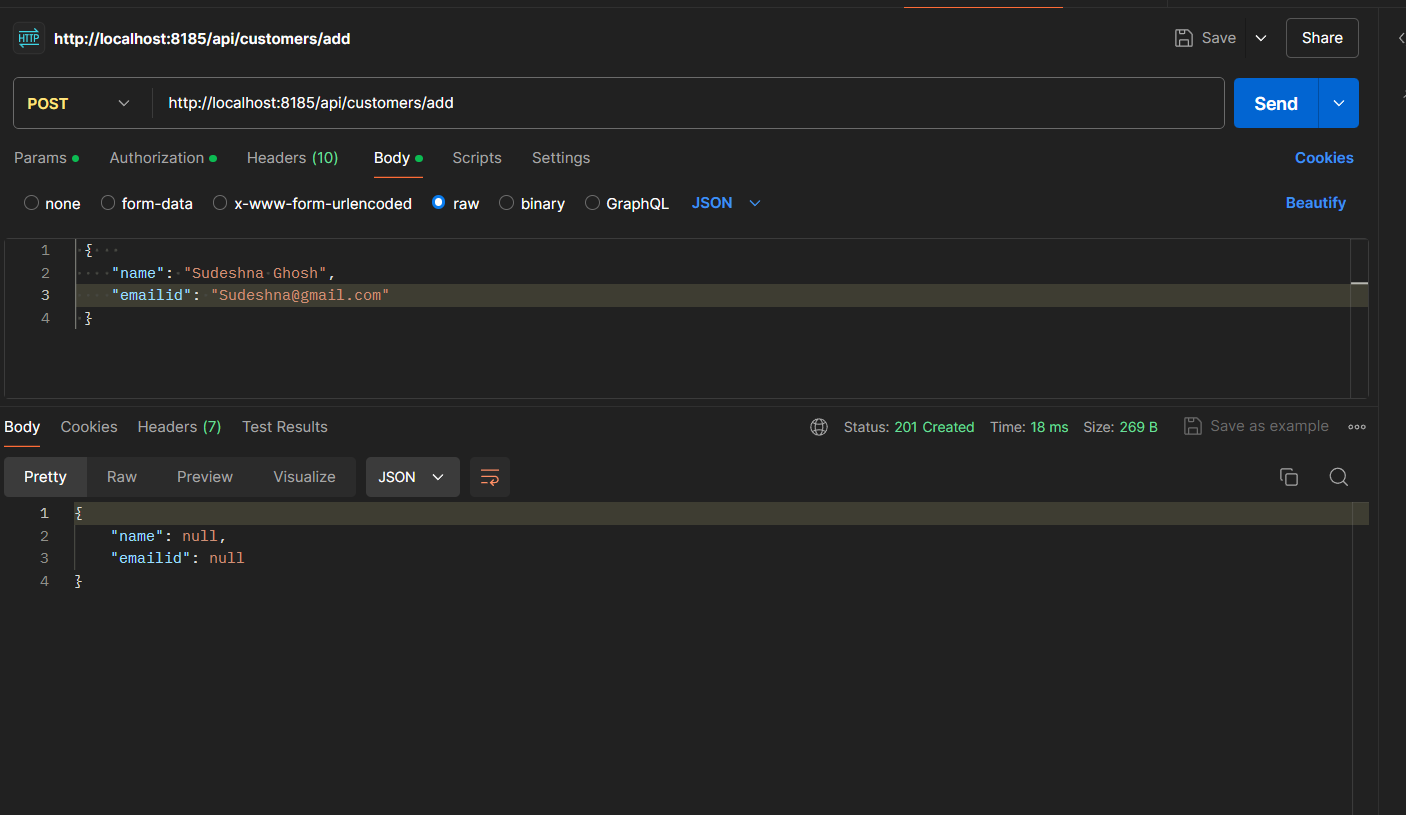
While adding book if the price is set as 0.00:



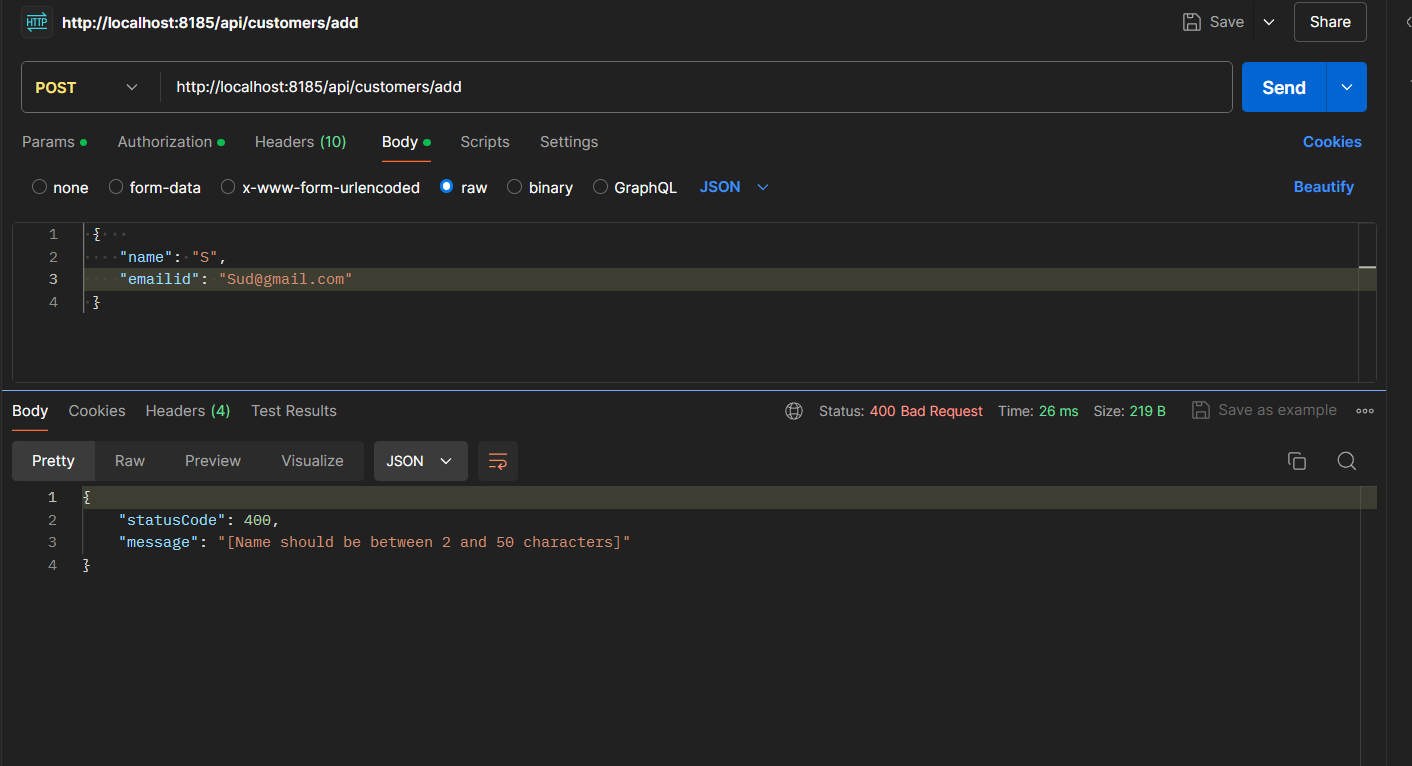
Adding another book:



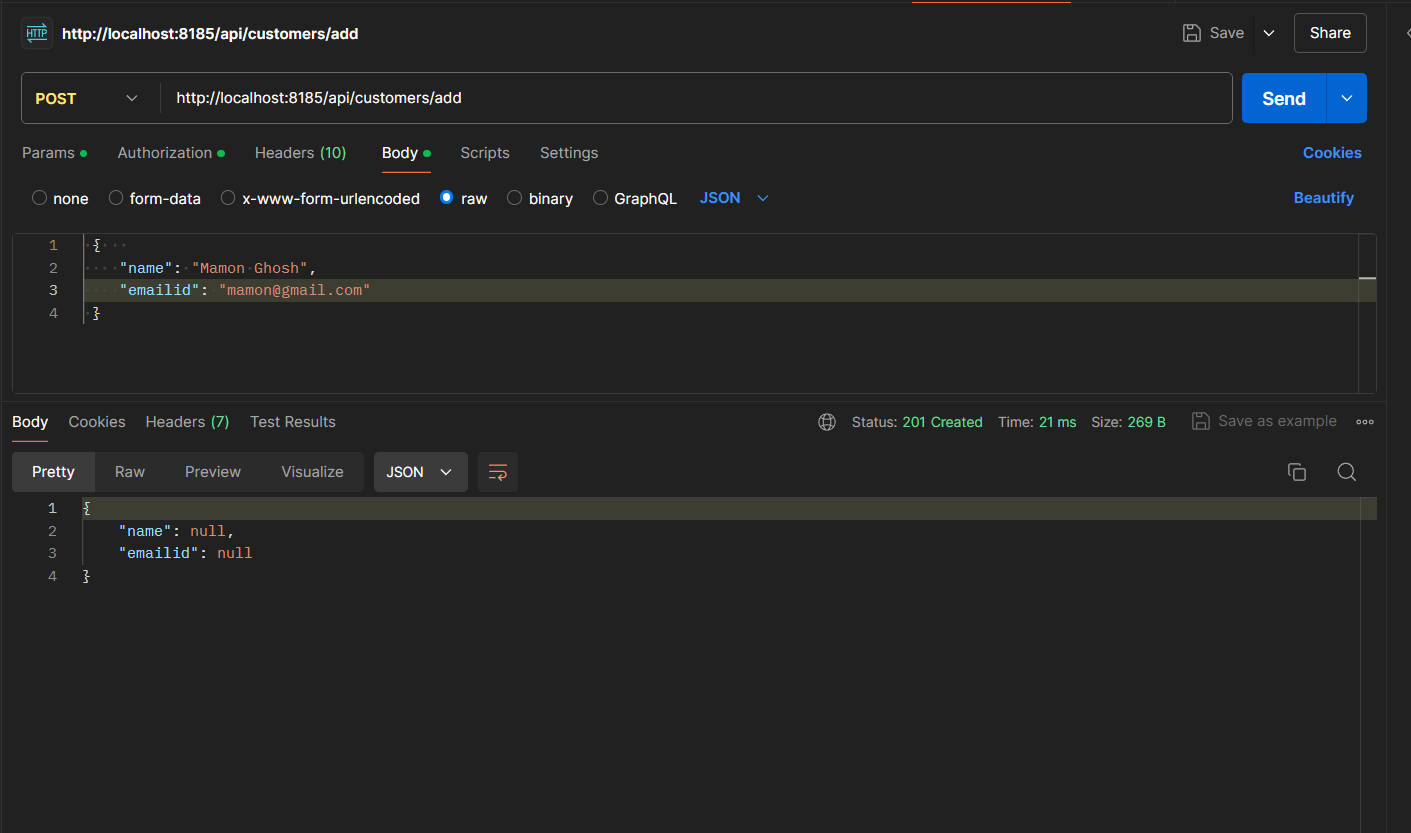
Adding a new Customer:



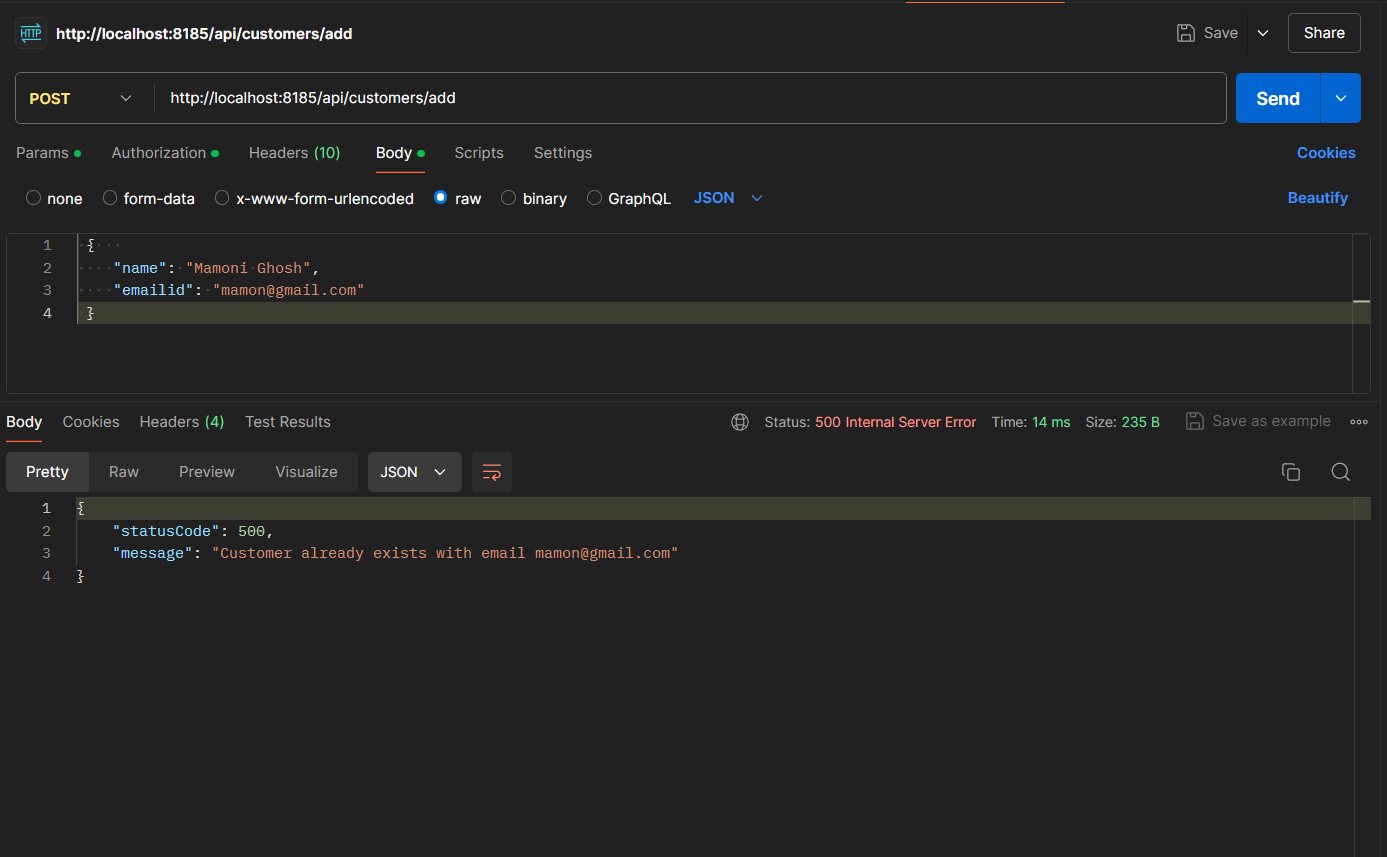
While adding is the name of the Customer is below 2 characters:



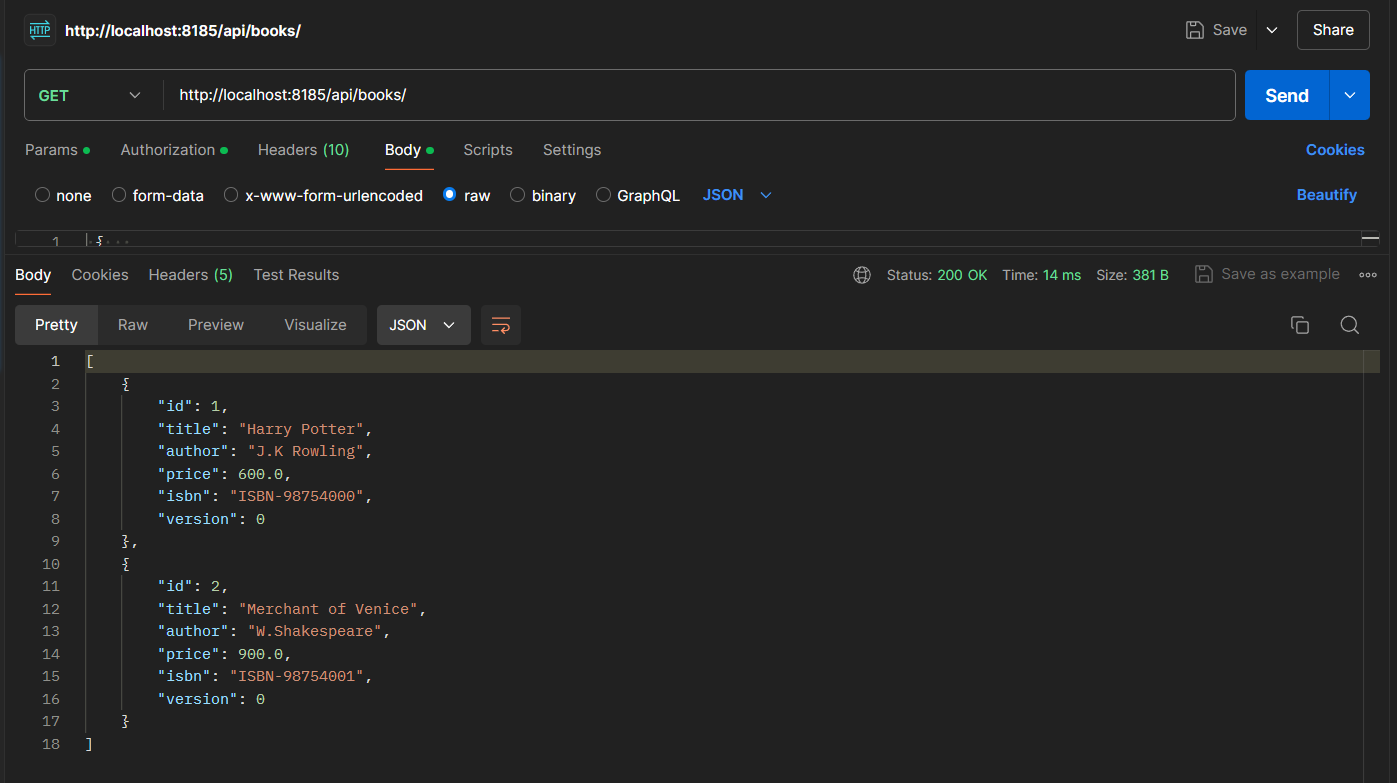
Adding a new customer:



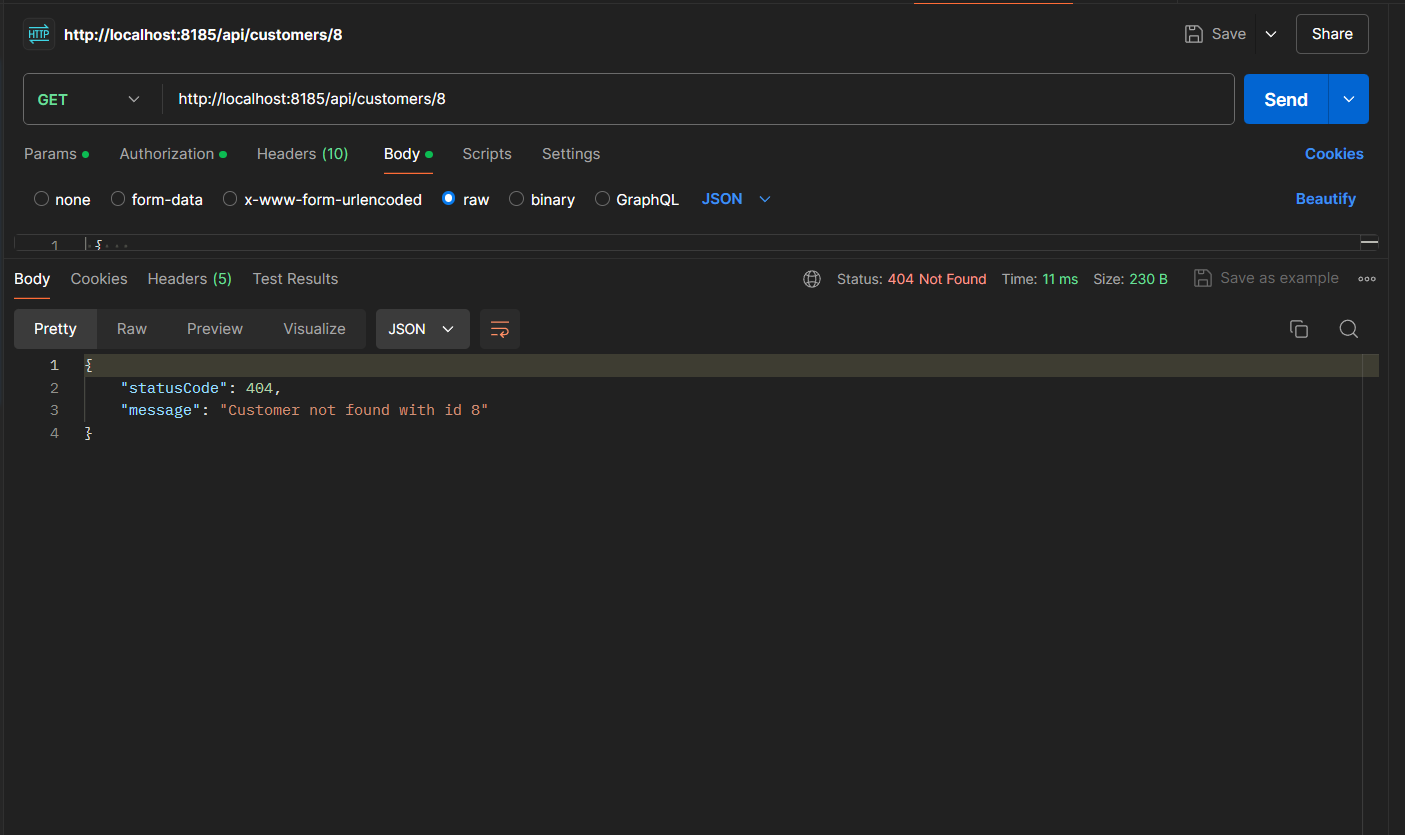
While adding customer if the email is same:



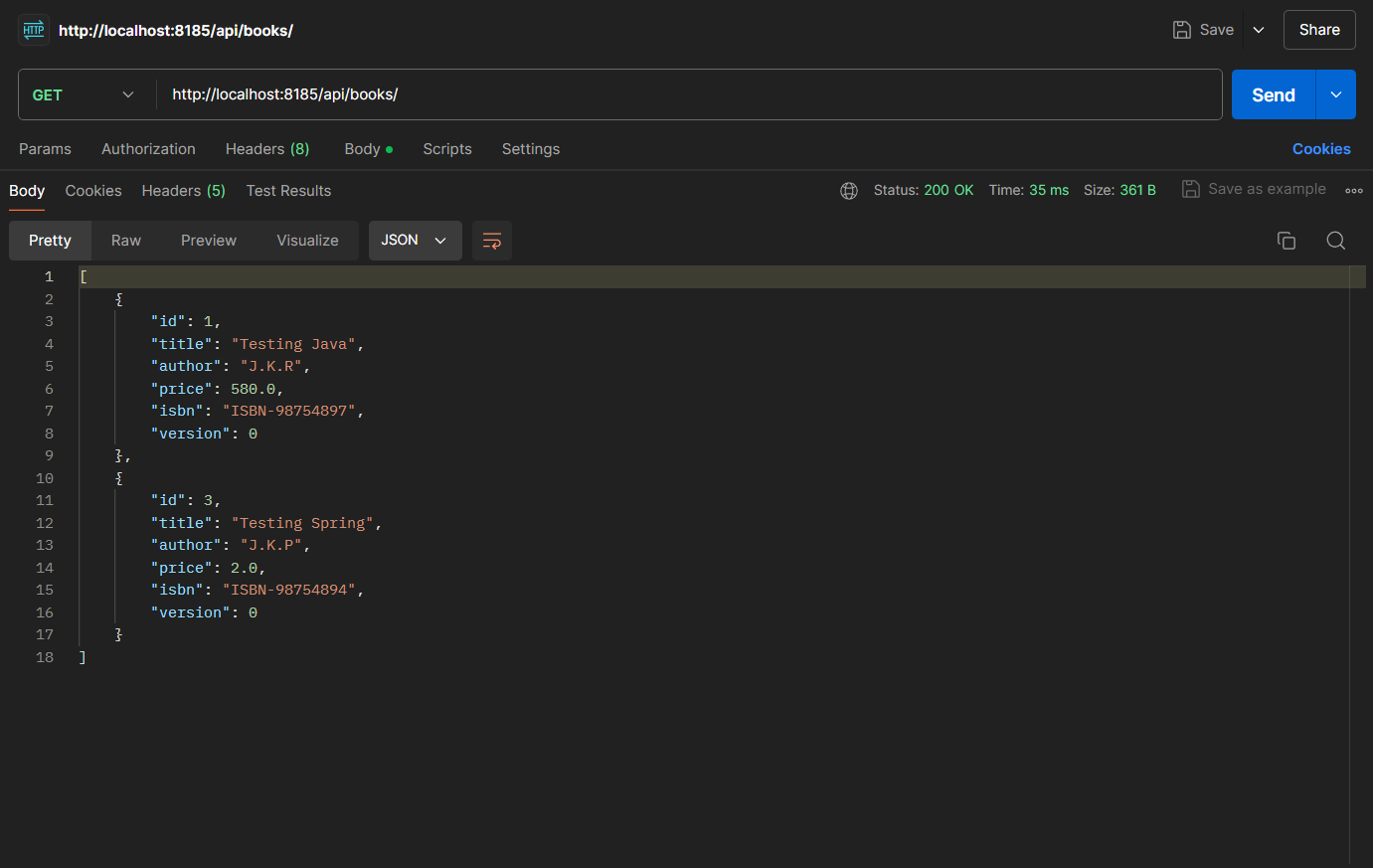
GET:



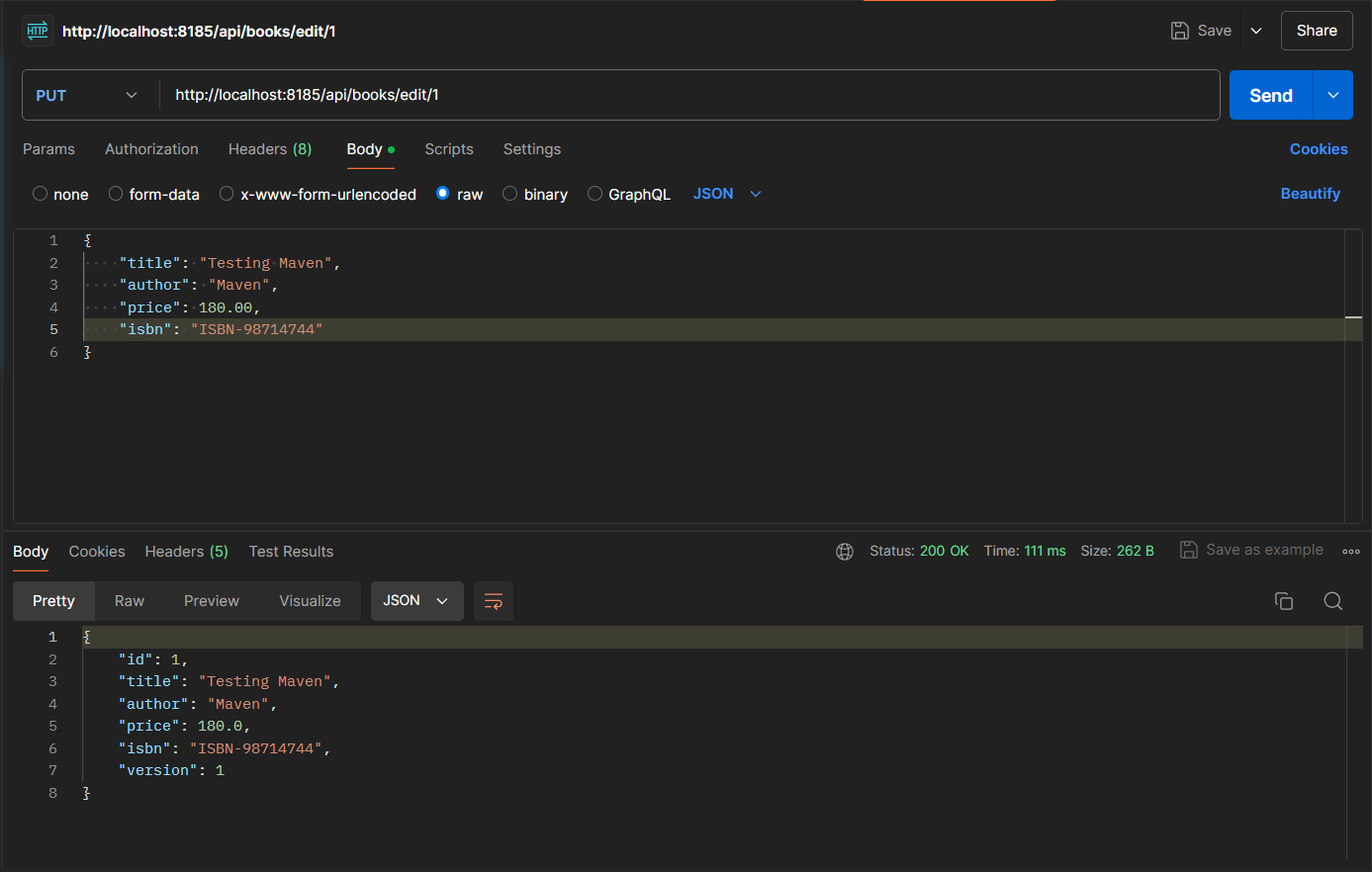
Customer not found:



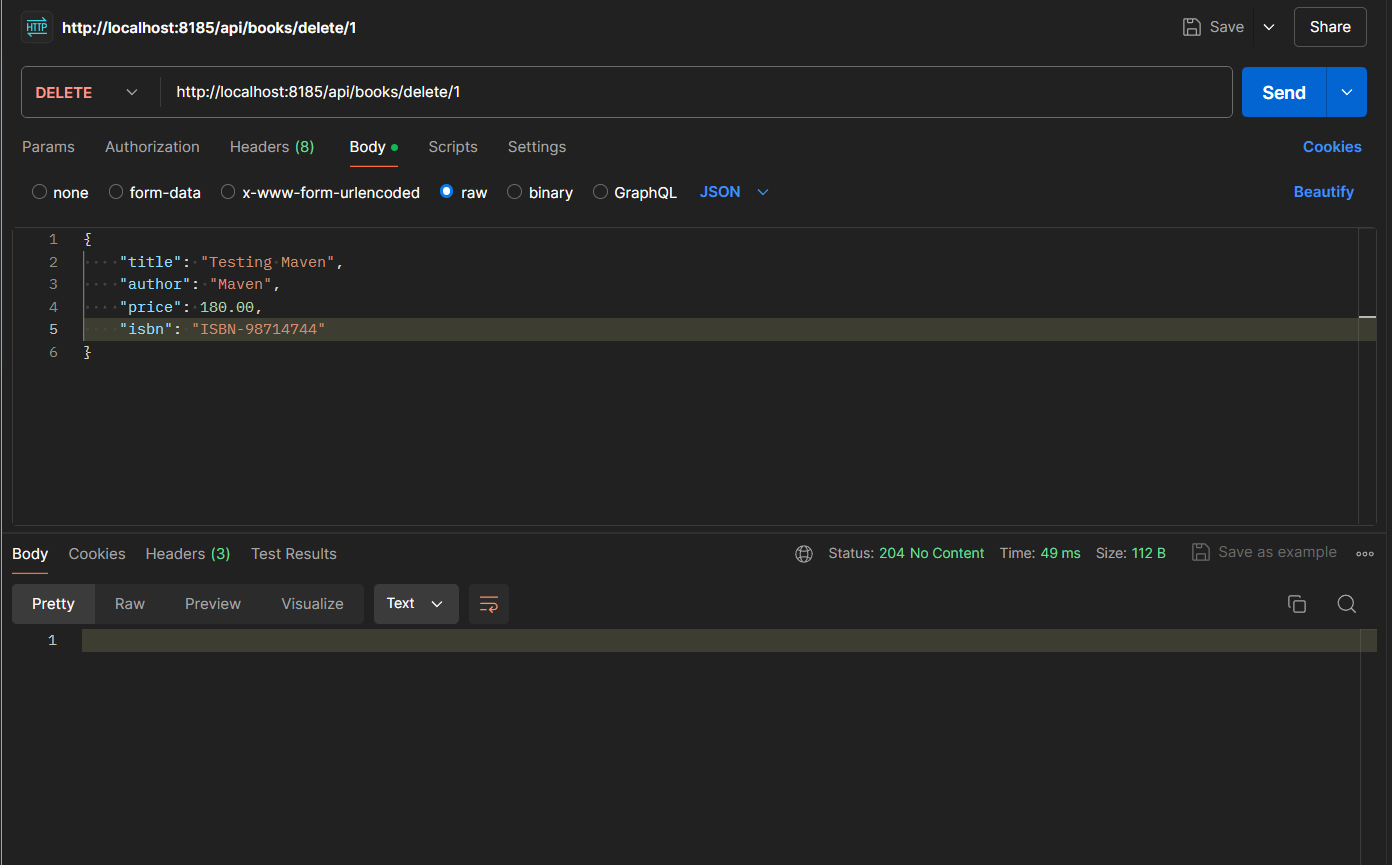
Removing and Adding other books for PUT and DELETE:



PUT:



DELETE:



Book not found:

