RESTful API for a Bookstore Management System - Documentation

Table of Contents

1. Adding a new book

Endpoint: /books

Method: POST

Request:

Body should contain JSON data with book details:

```
{
  "title": "The Life of John Doe",
  "author": "John Doe",
  "isbn": "123-4567890123",
  "price": “Rs. 123”,
  "quantity": 10
}
```

Authorization: Basic [base64-encoded-username:password]

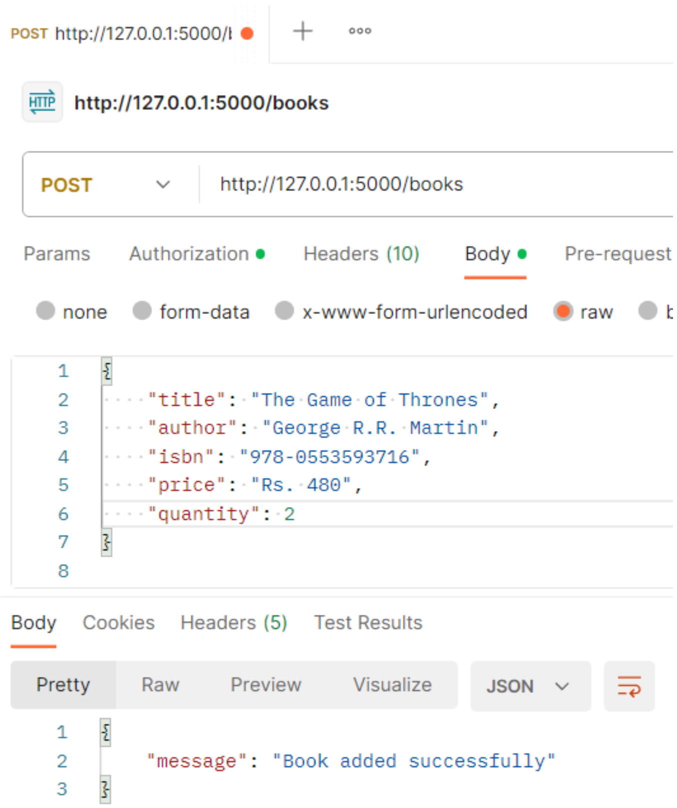
Response:

Successful response (HTTP status code 200):

{

"message": "Book added successfully"

}



http://127.0.0.1:5000/books

POST http://127.0.0.1:5000/books

```
{
    "title": "The Game of Thrones",
    "author": "George R.R. Martin",
    "isbn": "978-0553593716",
    "price": "Rs. 480",
    "quantity": 2
}

```

```
{
    "message": "Book added successfully"
}
```

Error response (HTTP status code 500):

{

"message": "Failed to add book"

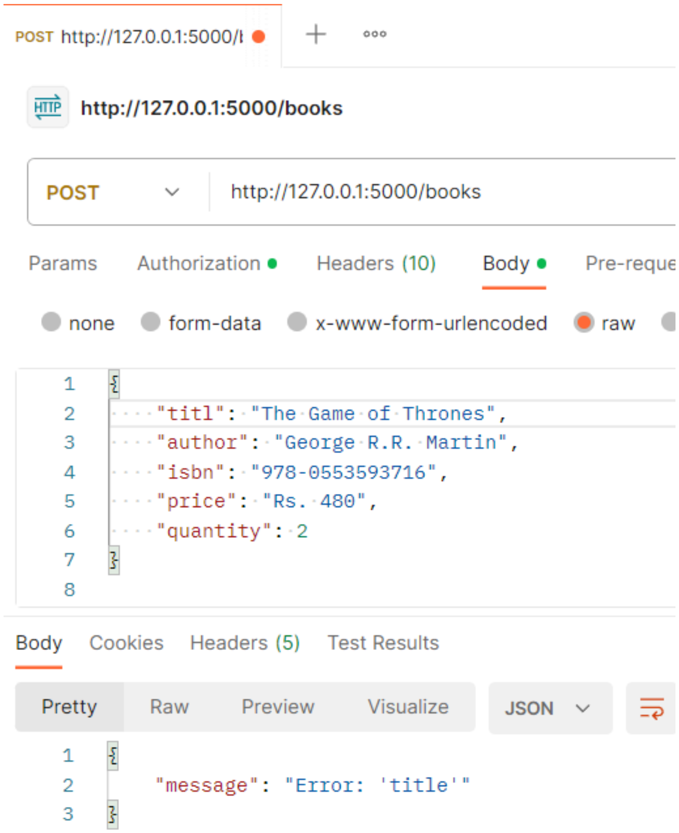
}

or

{

"message": "Error ‘{parameter}’"

}



http://127.0.0.1:5000/books

POST http://127.0.0.1:5000/books

Body ● raw

```
{
    "titl": "The Game of Thrones",
    "author": "George R.R. Martin",
    "isbn": "978-0553593716",
    "price": "Rs. 480",
    "quantity": 2
}
```

```
{
    "message": "Error: 'title'"
}
```

2. Retrieving all books

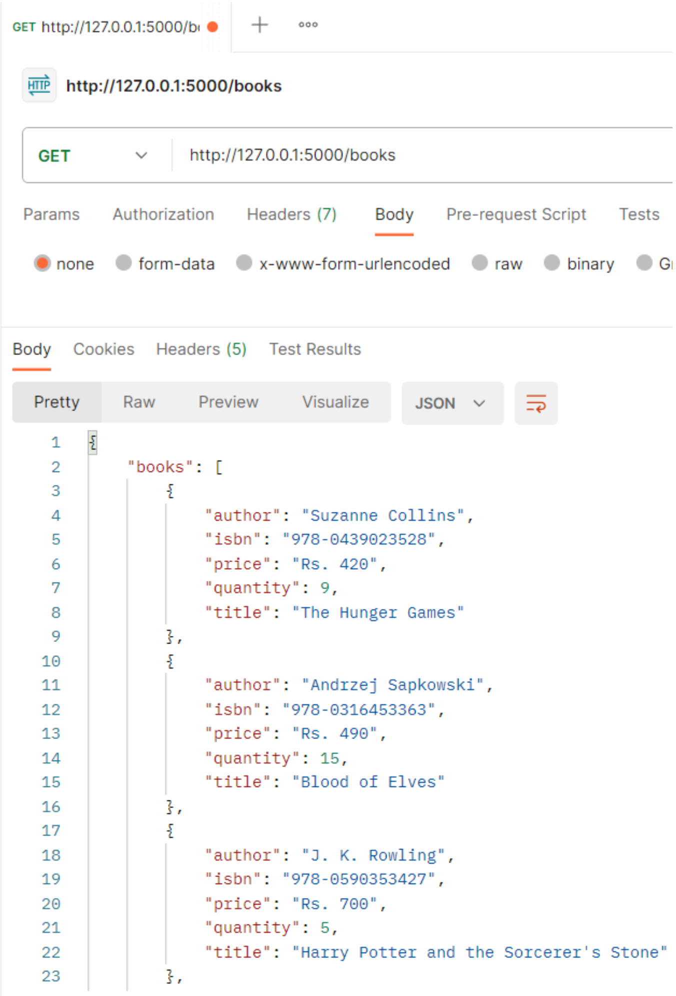
Endpoint: /books

Method: GET

Response:

Successful response (HTTP status code 200):

```
{
   "books": [
         {
          "title": "The Life of John Doe",
            "author": "John Doe",

            "isbn": "123-4567890123",

            "price": “Rs. 123”,

            "quantity": 10

        },

        // ... other books

    ]

}
```



http://127.0.0.1:5000/books

GET http://127.0.0.1:5000/books

Body: ● none

Body — Pretty — JSON

```
{
    "books": [
        {
            "author": "Suzanne Collins",
            "isbn": "978-0439023528",
            "price": "Rs. 420",
            "quantity": 9,
            "title": "The Hunger Games"
        },
        {
            "author": "Andrzej Sapkowski",
            "isbn": "978-0316453363",
            "price": "Rs. 490",
            "quantity": 15,
            "title": "Blood of Elves"
        },
        {
            "author": "J. K. Rowling",
            "isbn": "978-0590353427",
            "price": "Rs. 700",
            "quantity": 5,
            "title": "Harry Potter and the Sorcerer's Stone"
        },
```

3. Retrieving a specific book by ISBN

Endpoint: /books/{isbn}

Method: GET

Path Parameters:

isbn: ISBN number of the book.

Response:

Successful response (HTTP status code 200):

{

"title": "The Life of John Doe",

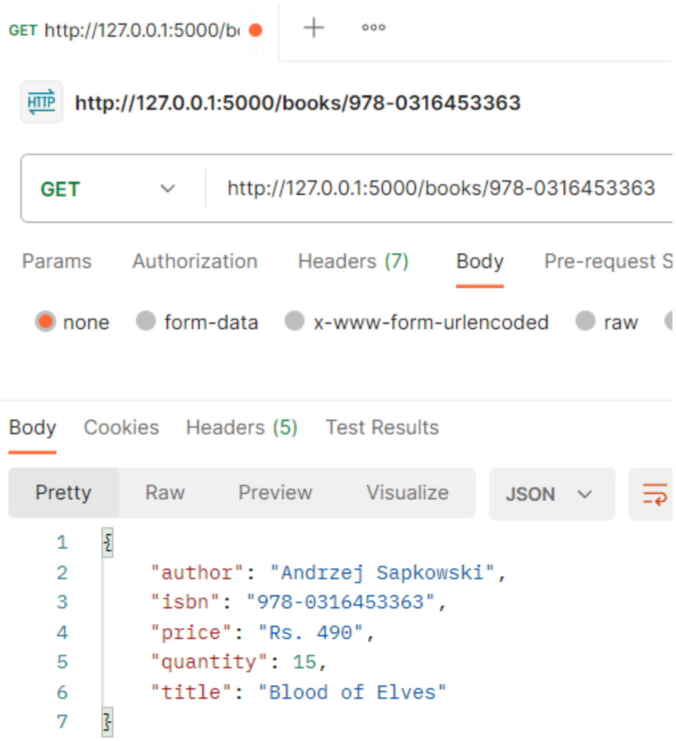
"author": "John Doe",

"isbn": "123-4567890123",

"price": “Rs. 123”,

"quantity": 10

}



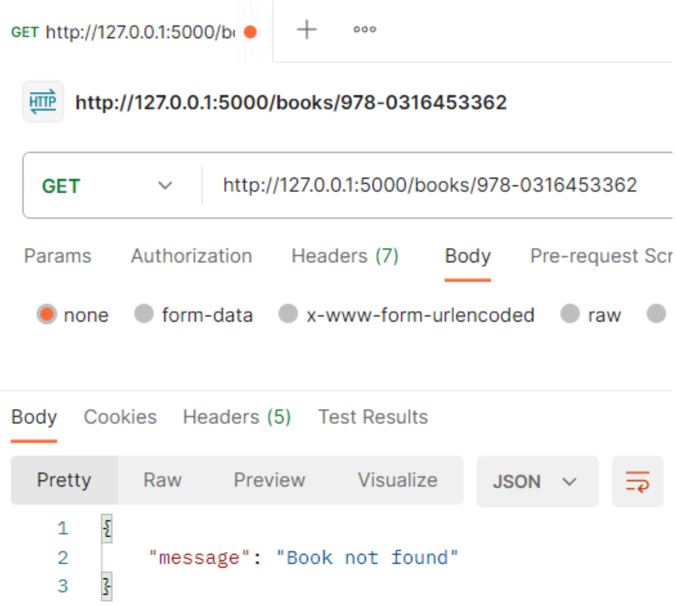
http://127.0.0.1:5000/books/978-0316453363

GET http://127.0.0.1:5000/books/978-0316453363

```
{
    "author": "Andrzej Sapkowski",
    "isbn": "978-0316453363",
    "price": "Rs. 490",
    "quantity": 15,
    "title": "Blood of Elves"
}
```

Error response (HTTP status code 404):

```
{
  "message": "Book not found"
}
```



GET http://127.0.0.1:5000/b

http://127.0.0.1:5000/books/978-0316453362

GET http://127.0.0.1:5000/books/978-0316453362

```
{
    "message": "Book not found"
}
```

4. Updating book details

Endpoint: /books/{isbn}

Method: PUT

Path Parameters:

isbn: ISBN number of the book.

Request:

Body should contain JSON data with updated book details.

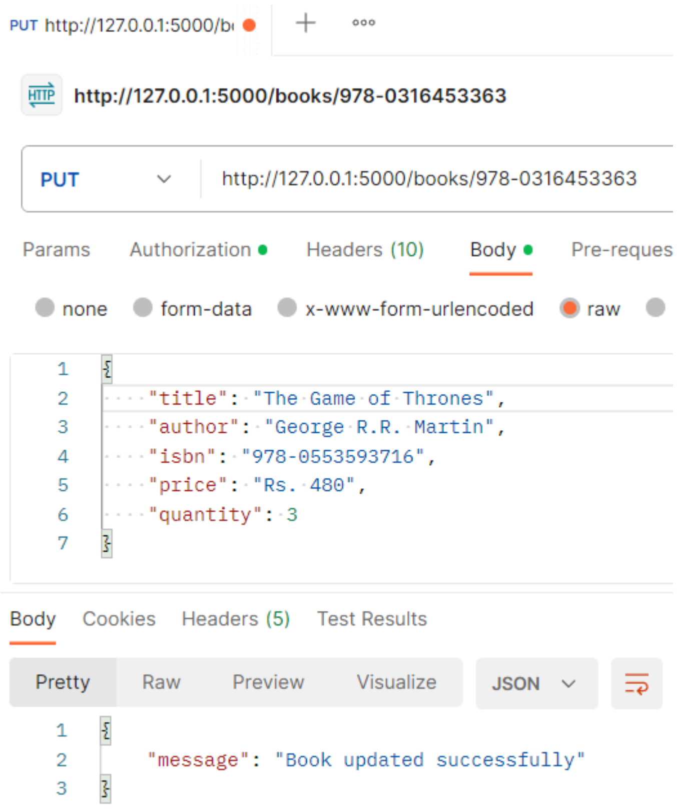
```
{
  "title": "The Life of John Doe",
  "author": "John Doe",
  "isbn": "123-4567890123",
  "price": “Rs. 123”,
  "quantity": 5
}
```

Authorization: Basic [base64-encoded-username:password]

Response:

Successful response (HTTP status code 200):

```
{
  "message": "Book updated successfully"
}
```



PUT http://127.0.0.1:5000/books/978-0316453363

```
{
    "title": "The Game of Thrones",
    "author": "George R.R. Martin",
    "isbn": "978-0553593716",
    "price": "Rs. 480",
    "quantity": 3
}
```

```
{
    "message": "Book updated successfully"
}
```

5. Deleting a book

Endpoint: /books/{isbn}

Method: DELETE

Path Parameters:

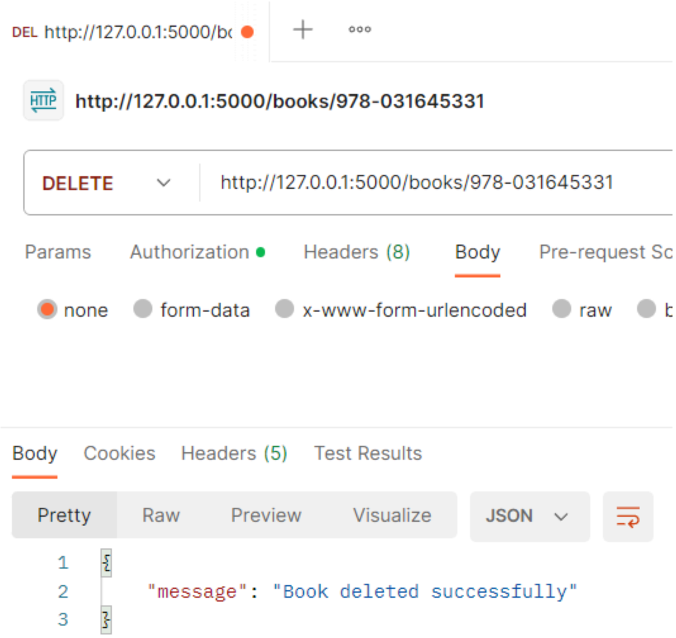
isbn: ISBN number of the book.

Authorization: Basic [base64-encoded-username:password]

Response:

Successful response (HTTP status code 200):

```
{
  "message": "Book deleted successfully"
}
```



DEL http://127.0.0.1:5000/bo

http://127.0.0.1:5000/books/978-031645331

DELETE http://127.0.0.1:5000/books/978-031645331

```
{
    "message": "Book deleted successfully"
}
```