

RESTful API for a Bookstore Management System - Documentation

Table of Contents

Sl. No.	Title	Page No.
1	Adding a new book	1
2	Retrieving all books	3
3	Retrieving a specific book by ISBN	5
4	Updating book details	6
5	Deleting a book	7

1. Adding a new book

Endpoint: /books

Method: POST

Request:

Body should contain JSON data with book details:

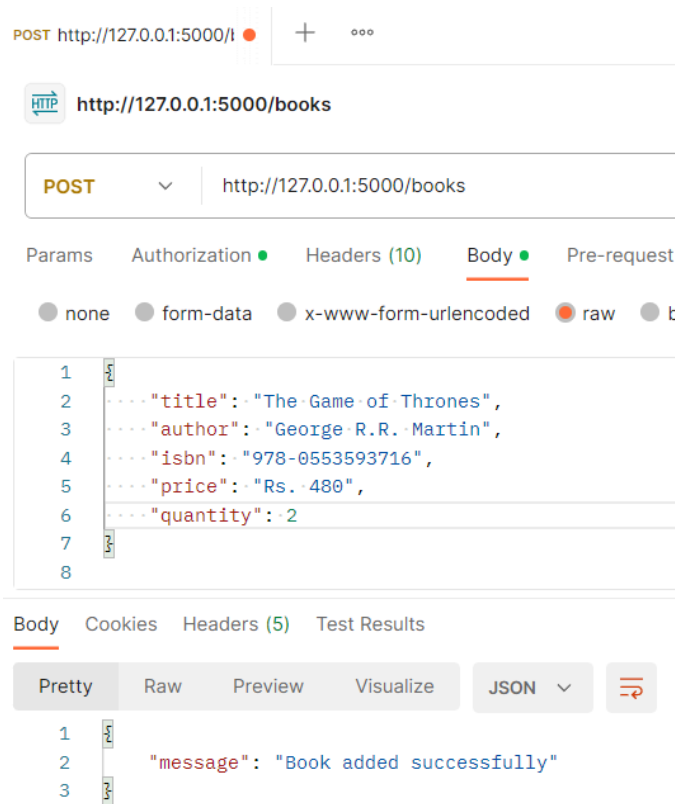
```
{  
  "title": "The Life of John Doe",  
  "author": "John Doe",  
  "isbn": "123-4567890123",  
  "price": "Rs. 123",  
  "quantity": 10  
}
```

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

Response:

Successful response (HTTP status code 200):

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



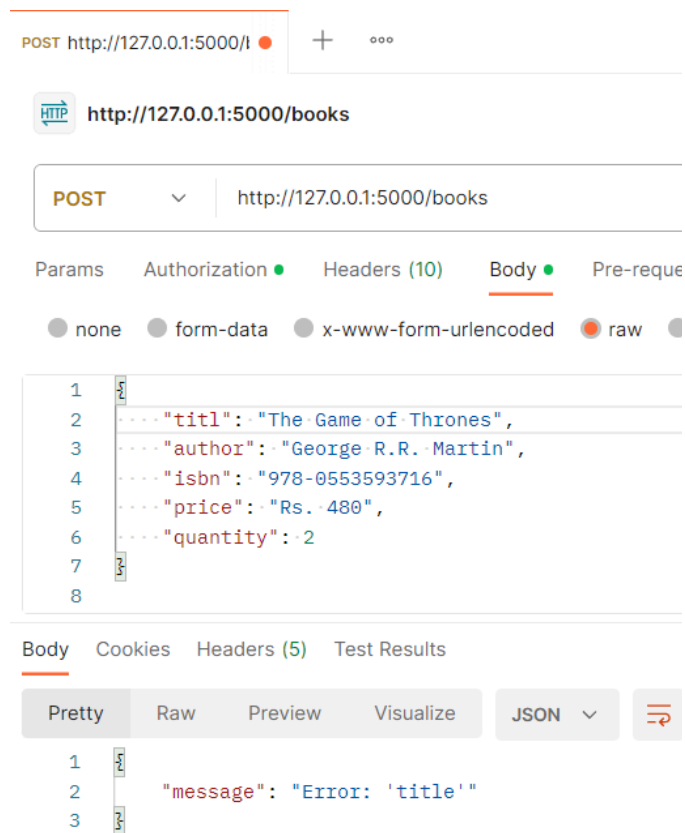
Error response (HTTP status code 500):

```
{  
  "message": "Failed to add book"  
}
```

or

```
{
```

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



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
]
}

```

The screenshot shows a web browser's developer tools interface. At the top, a GET request is shown for the URL `http://127.0.0.1:5000/books`. Below this, the 'Body' tab is selected, displaying the response in JSON format. The JSON response is a list of three books, each with fields for author, isbn, price, quantity, and title. The books are 'The Hunger Games' by Suzanne Collins, 'Blood of Elves' by Andrzej Sapkowski, and 'Harry Potter and the Sorcerer's Stone' by J. K. Rowling.

```

{
  "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

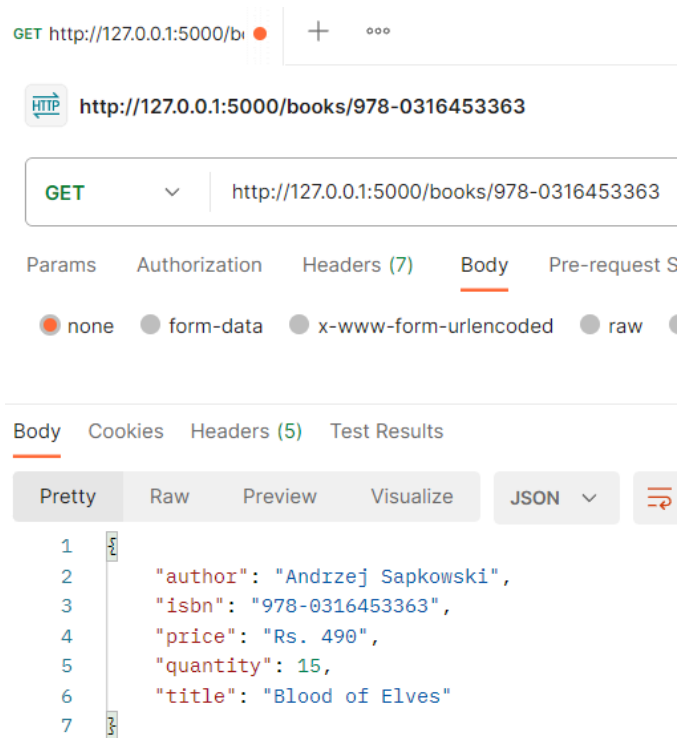
Path Parameters:

isbn: ISBN number of the book.

Response:

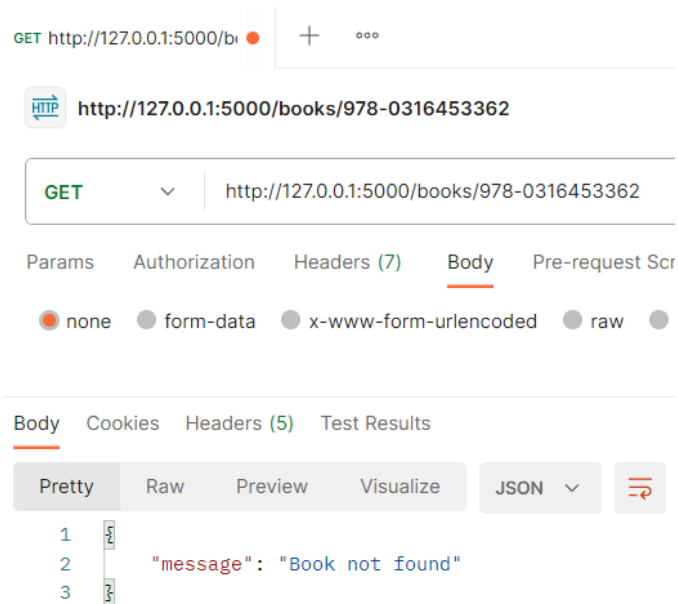
Successful response (HTTP status code 200):

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



Error response (HTTP status code 404):

```
{  
  "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",  
}
```

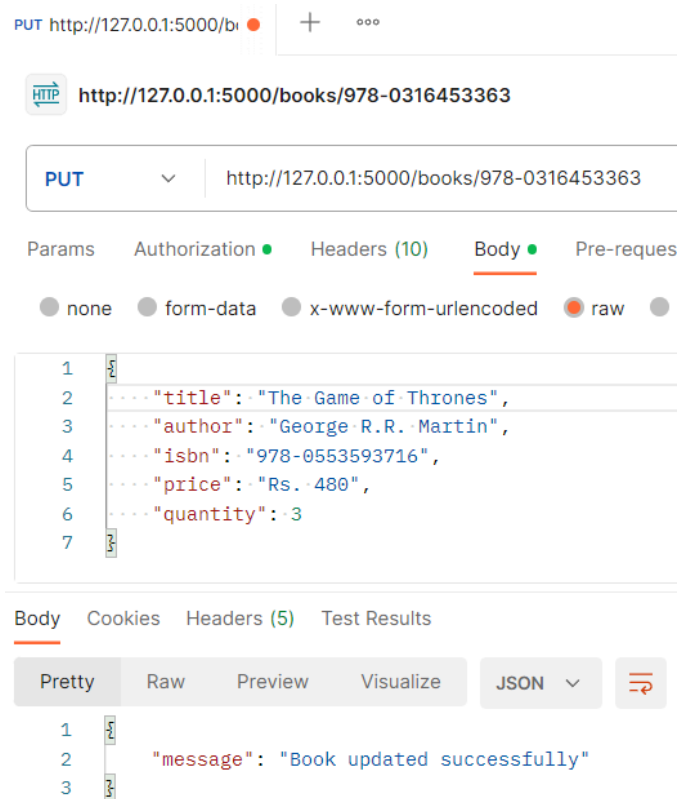
```
"isbn": "123-4567890123",  
  
"price": "Rs. 123",  
  
"quantity": 5  
}
```

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

Response:

Successful response (HTTP status code 200):

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



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"  
}
```

