RESTful API for a Bookstore Management System - Documentation

Table of Contents

SI. 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]

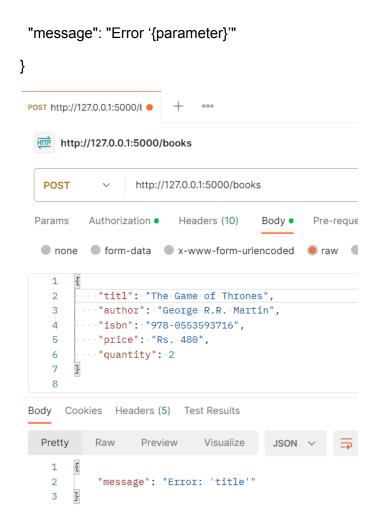
Response:

```
Successful response (HTTP status code 200):
```

```
{
 "message": "Book added successfully"
}
 POST http://127.0.0.1:5000/i
  http://127.0.0.1:5000/books
   POST
                   http://127.0.0.1:5000/books
  Params
           Authorization •
                          Headers (10)
                                         Body •
                                                  Pre-request
   none form-data x-www-form-urlencoded
                                                 naw t
     1
     2
          ··· "title": "The Game of Thrones",
     3
          "author": "George R.R. Martin",
     4
            "isbn": "978-0553593716",
            "price": "Rs. 480",
     5
             "quantity": 2
     6
     7
     8
 Body Cookies Headers (5) Test Results
   Pretty
            Raw
                    Preview
                               Visualize
             "message": "Book added successfully"
     2
```

Error response (HTTP status code 500):

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



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
  ]
}
 GET http://127.0.0.1:5000/bi • + •••
  http://127.0.0.1:5000/books
                   http://127.0.0.1:5000/books
   GET
  Params
           Authorization
                        Headers (7)
                                      Body
                                             Pre-request Script
   ● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● G
 Body Cookies Headers (5) Test Results
   Pretty
                    Preview
                               Visualize
                                          JSON V
     1
     2
             "books": [
     3
                     "author": "Suzanne Collins",
                     "isbn": "978-0439023528",
     5
                     "price": "Rs. 420",
                     "quantity": 9,
                     "title": "The Hunger Games"
     8
     9
    10
    11
                     "author": "Andrzej Sapkowski",
                     "isbn": "978-0316453363",
    12
    13
                     "price": "Rs. 490",
                     "quantity": 15,
    14
    15
                     "title": "Blood of Elves"
    16
    17
                     "author": "J. K. Rowling",
    18
                     "isbn": "978-0590353427",
    19
                     "price": "Rs. 700",
    20
                     "quantity": 5,
    21
    22
                     "title": "Harry Potter and the Sorcerer's Stone"
    23
```

3. Retrieving a specific book by ISBN

Endpoint: /books/{isbn}

Method: GET

Path Parameters:

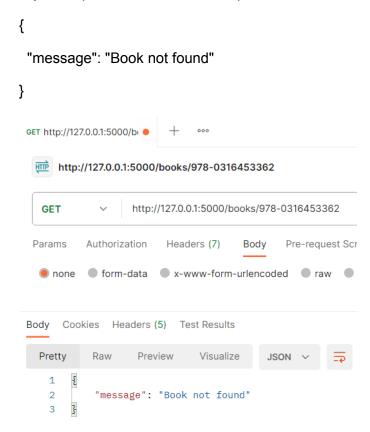
isbn: ISBN number of the book.

Response:

Successful response (HTTP status code 200):

```
{
 "title": "The Life of John Doe",
 "author": "John Doe",
 "isbn": "123-4567890123",
 "price": "Rs. 123",
 "quantity": 10
}
 GET http://127.0.0.1:5000/bi
  http://127.0.0.1:5000/books/978-0316453363
   GET
                   http://127.0.0.1:5000/books/978-0316453363
  Params
           Authorization
                          Headers (7)
                                       Body
                                               Pre-request S
   none  form-data  x-www-form-urlencoded  raw  
      Cookies Headers (5) Test Results
 Body
   Pretty
                    Preview
                                Visualize
     1
     2
             "author": "Andrzej Sapkowski",
     3
             "isbn": "978-0316453363",
             "price": "Rs. 490",
     4
             "quantity": 15,
             "title": "Blood of Elves"
     6
```

Error response (HTTP status code 404):



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/b( •
                  http://127.0.0.1:5000/books/978-0316453363
                   PUT
                                  http://127.0.0.1:5000/books/978-0316453363
                  Params
                           Authorization •
                                         Headers (10)
                                                       Body •
                                                                Pre-reques
                   none form-data x-www-form-urlencoded raw
                    1
                            "title": · "The · Game · of · Thrones",
                    2
                           "author": "George R.R. Martin",
                    3
                         ...."isbn": "978-0553593716",
                         ····"price": "Rs. 480",
                          ··· "quantity": 3
                 Body Cookies Headers (5) Test Results
```

Visualize

"message": "Book updated successfully"

5. Deleting a book

Endpoint: /books/{isbn}

Pretty

1

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/bc http://127.0.0.1:5000/books/978-031645331 DELETE http://127.0.0.1:5000/books/978-031645331 Params Authorization • Headers (8) Body Pre-request Sc form-data x-www-form-urlencoded raw b Body Cookies Headers (5) Test Results Visualize Pretty Raw Preview JSON 2 "message": "Book deleted successfully"