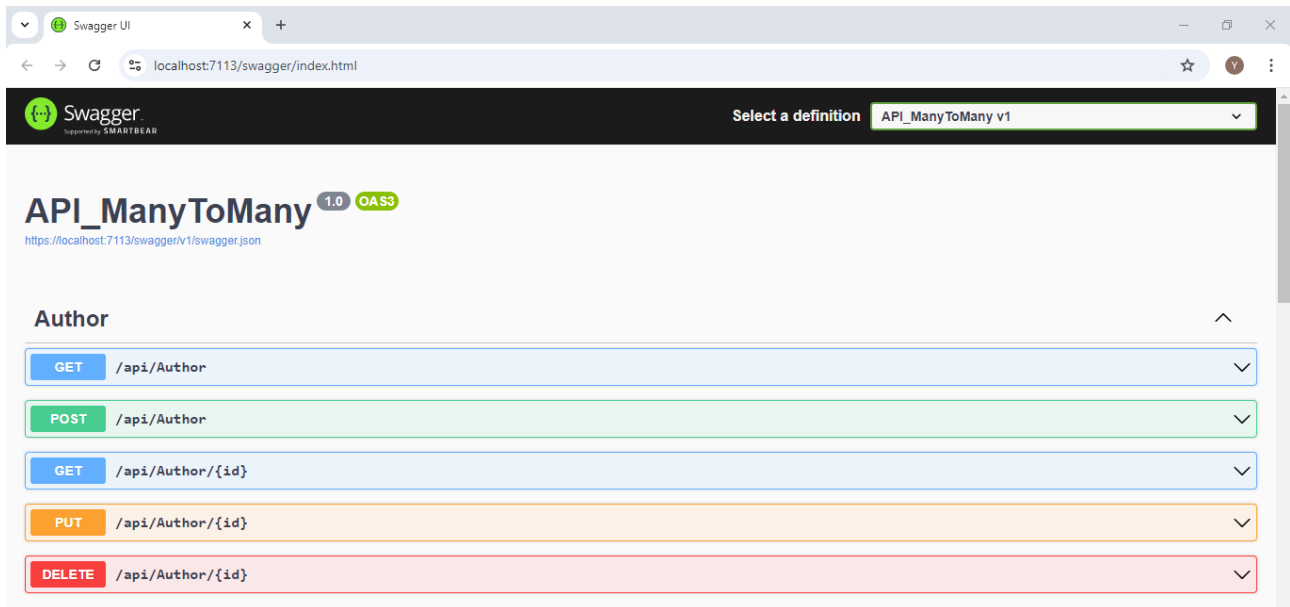


DATE : 05/09/2024

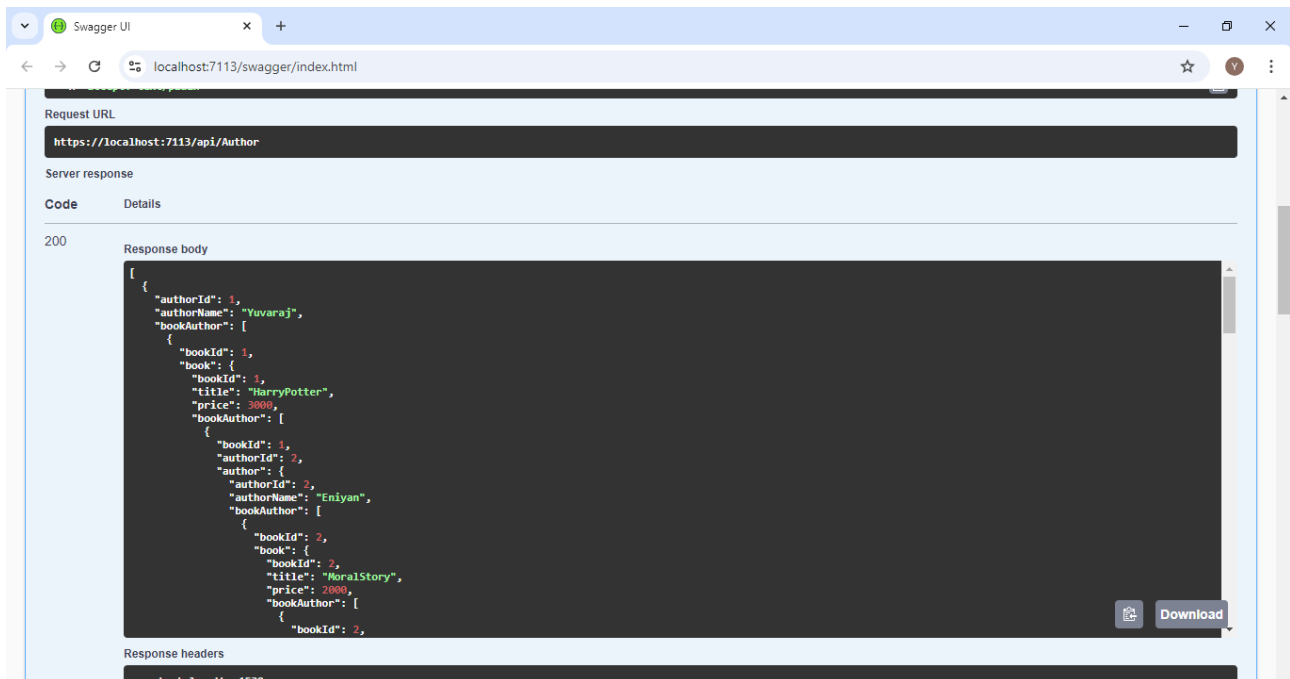
API – Repository Pattern

Many to Many Relation:

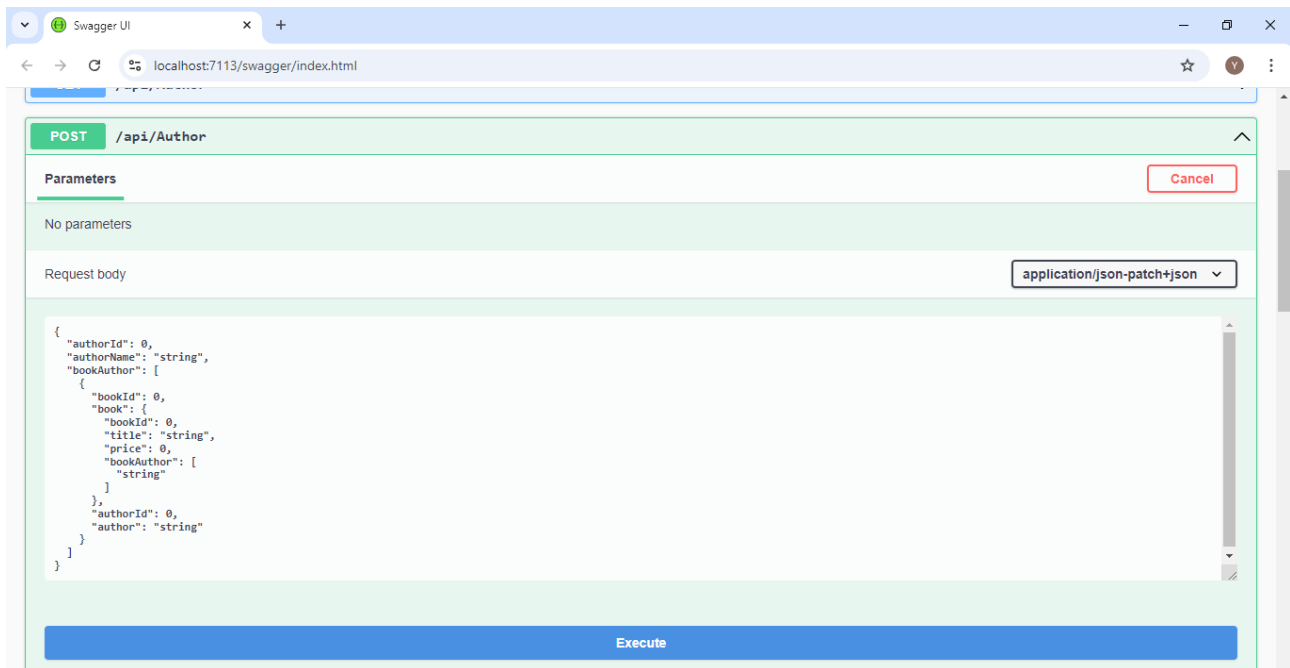
Author Controller :



Get :



Post :

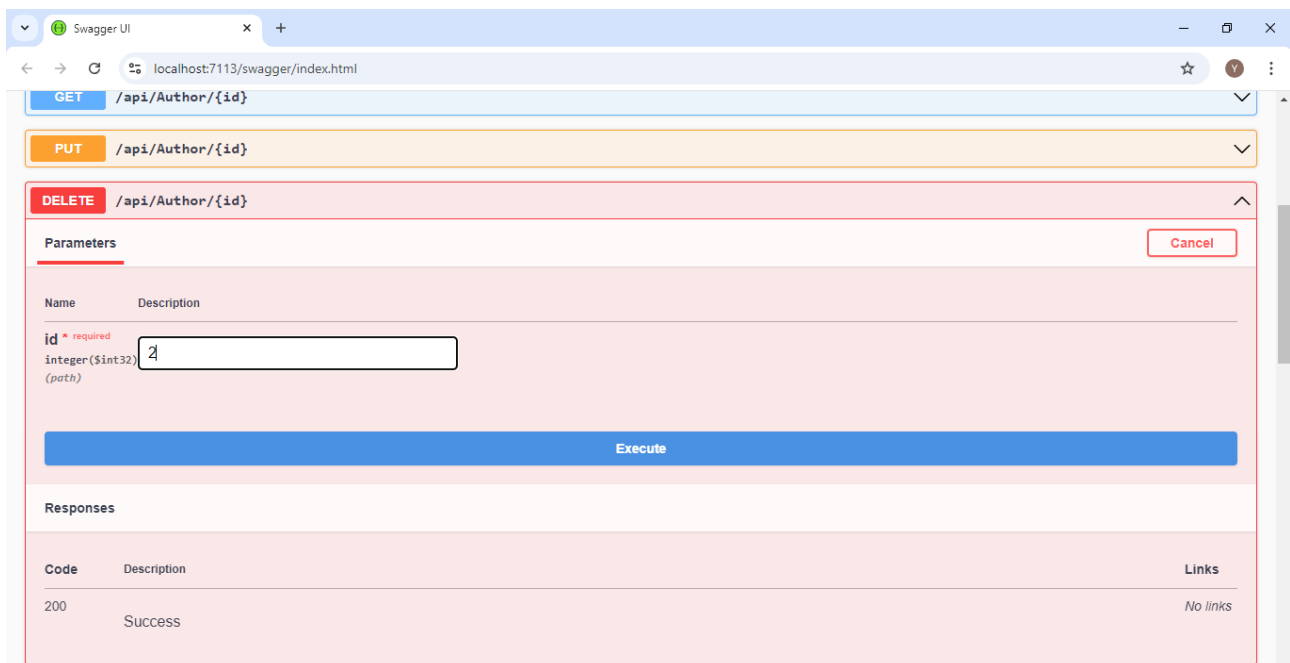


The image shows the Swagger UI interface for a POST request to the endpoint `/api/Author`. The interface is displayed in a web browser window with the address `localhost:7113/swagger/index.html`. The request method is `POST`, and the media type is set to `application/json-patch+json`. The request body is a JSON object with the following structure:

```
{
  "authorId": 0,
  "authorName": "string",
  "bookAuthor": [
    {
      "bookId": 0,
      "book": {
        "bookId": 0,
        "title": "string",
        "price": 0,
        "bookAuthor": [
          "string"
        ]
      },
      "authorId": 0,
      "author": "string"
    }
  ]
}
```

At the bottom of the interface, there is a blue `Execute` button.

Delete:



The image shows the Swagger UI interface for a DELETE request to the endpoint `/api/Author/{id}`. The interface is displayed in a web browser window with the address `localhost:7113/swagger/index.html`. The request method is `DELETE`. The parameters section shows a required integer parameter `id` with a value of `2` entered in the input field. The responses section shows a 200 status code with a description of "Success".

Name	Description
<code>id</code> * required <code>integer(\$int32)</code> (path)	2

At the bottom of the interface, there is a blue `Execute` button.

Code	Description	Links
200	Success	No links

Put:

The image shows the Swagger UI interface for the PUT endpoint `/api/Author/{id}`. The interface is displayed in a web browser window with the title "Swagger UI" and the URL `localhost:7113/swagger/index.html`. The endpoint is highlighted in orange. Below the endpoint name, there is a "Parameters" section with a "Cancel" button. The parameters table has two columns: "Name" and "Description". The first parameter is `id`, which is required, of type `integer($int32)`, and is a path parameter. Below the parameters section, there is a "Request body" section with a dropdown menu set to `application/json-patch+json`. The request body is a JSON object representing a patch operation on an author's books.

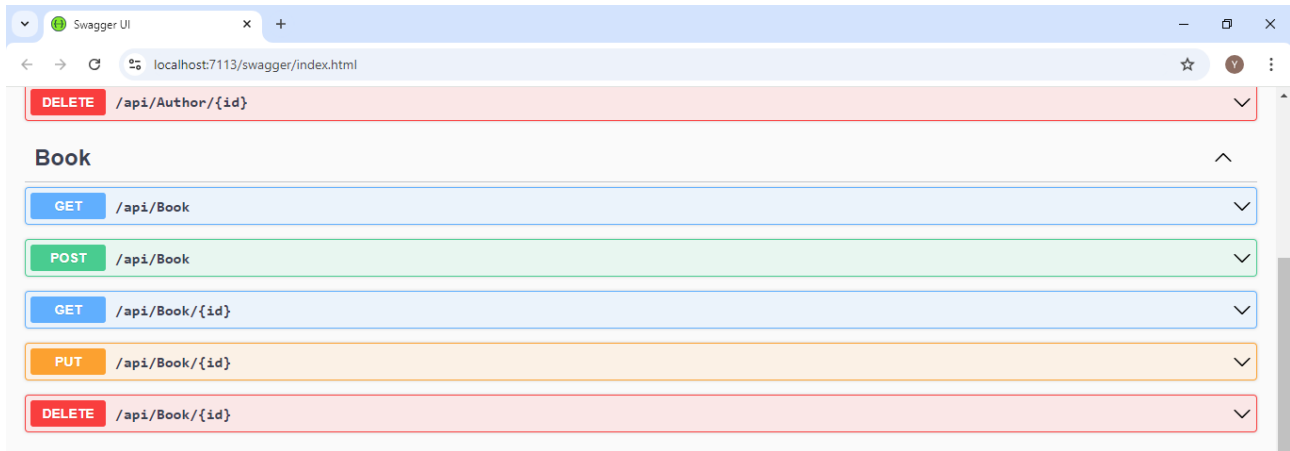
```
{
  "authorId": 0,
  "authorName": "string",
  "bookAuthor": [
    {
      "bookId": 0,
      "book": {
        "bookId": 0,
        "title": "string",
        "price": 0,
        "bookAuthor": [
          "string"
        ]
      }
    },
    "authorId": 0,
    "author": "string"
  ]
}
```

Get By ID:

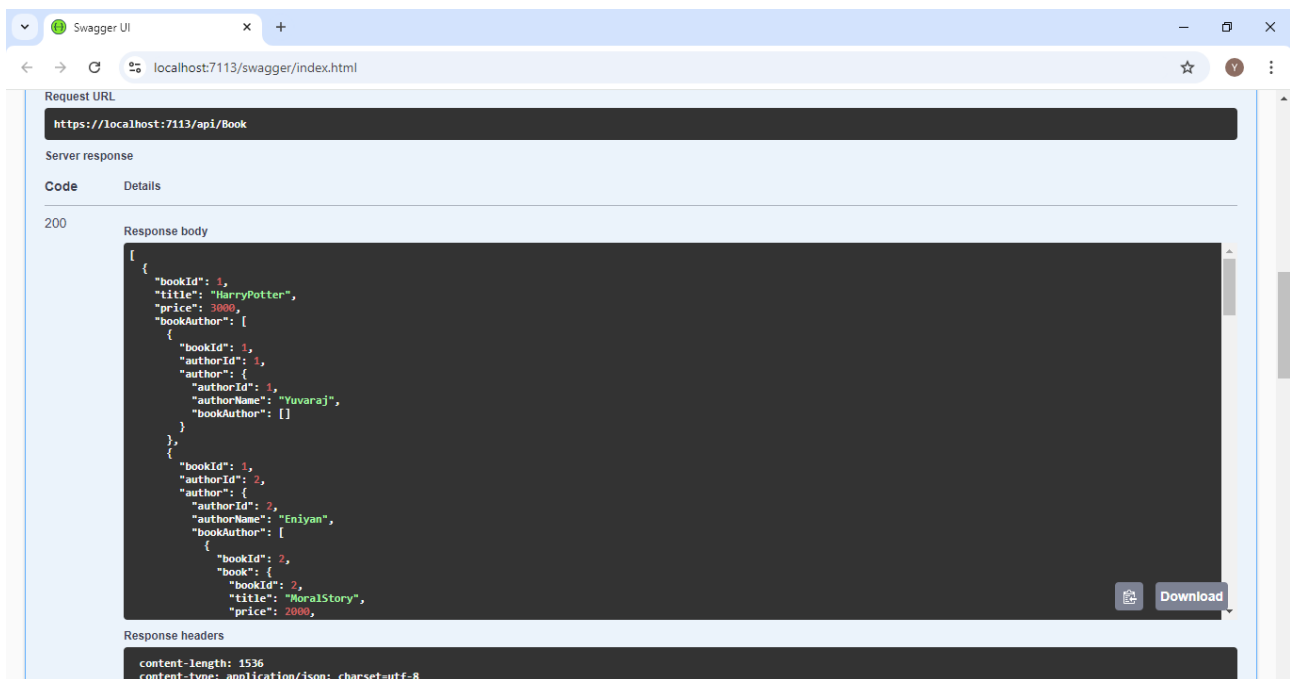
The image shows the Swagger UI interface for the GET endpoint `/api/Author/{id}`. The interface is displayed in a web browser window with the title "Swagger UI" and the URL `localhost:7113/swagger/index.html`. The endpoint is highlighted in blue. Below the endpoint name, there is a "Parameters" section with a "Cancel" button. The parameters table has two columns: "Name" and "Description". The first parameter is `id`, which is required, of type `integer($int32)`, and is a path parameter. The value `2` is entered in the input field for the `id` parameter. Below the parameters section, there is a blue "Execute" button. Below the "Execute" button, there is a "Responses" section. The responses table has three columns: "Code", "Description", and "Links". The first response is `200`, with the description "Success" and "No links".

Code	Description	Links
200	Success	No links

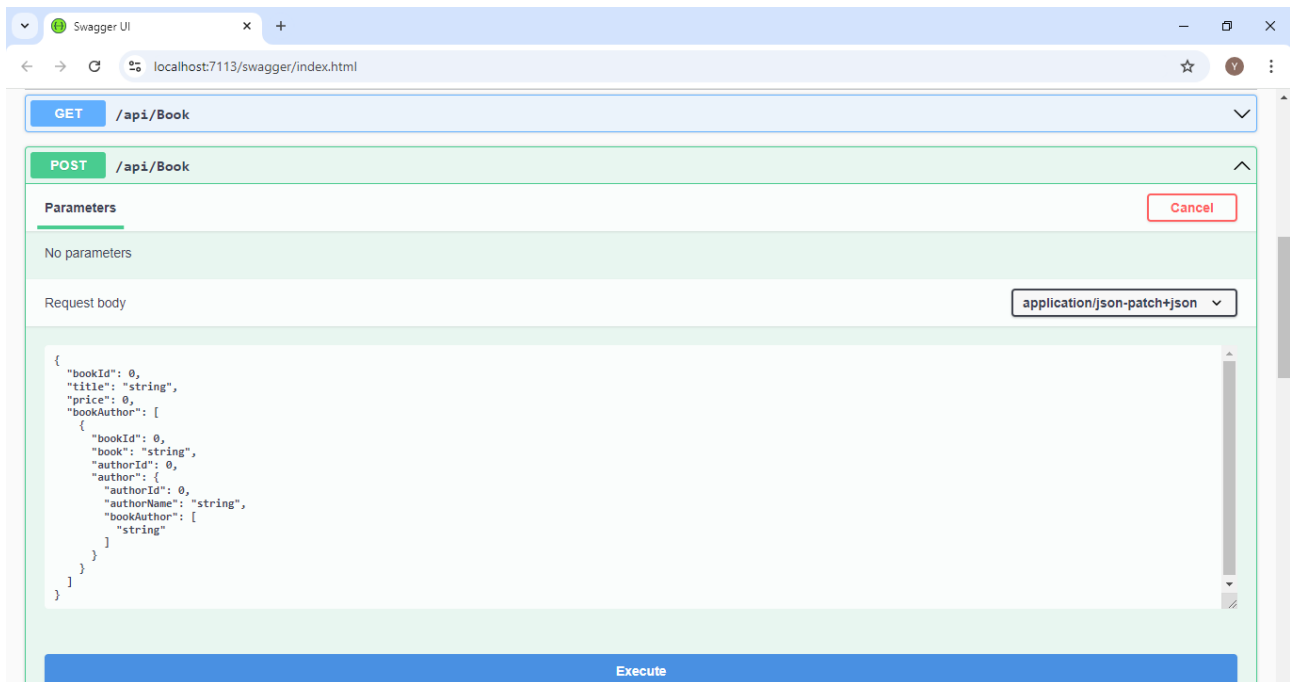
Book Controller:



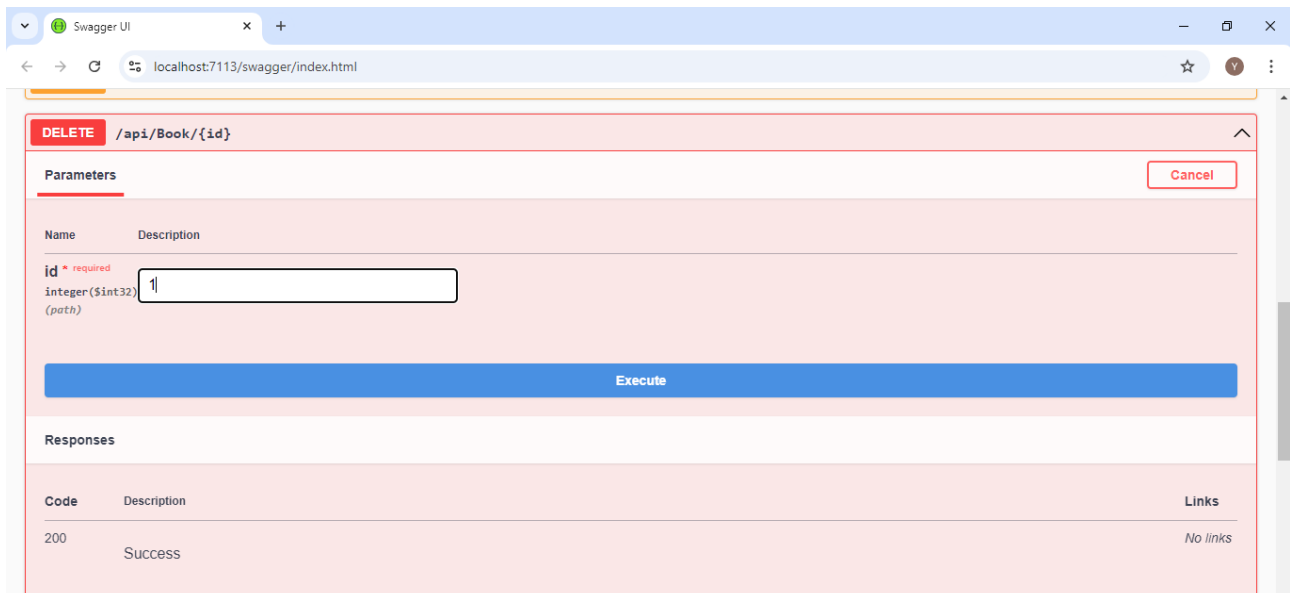
Get:



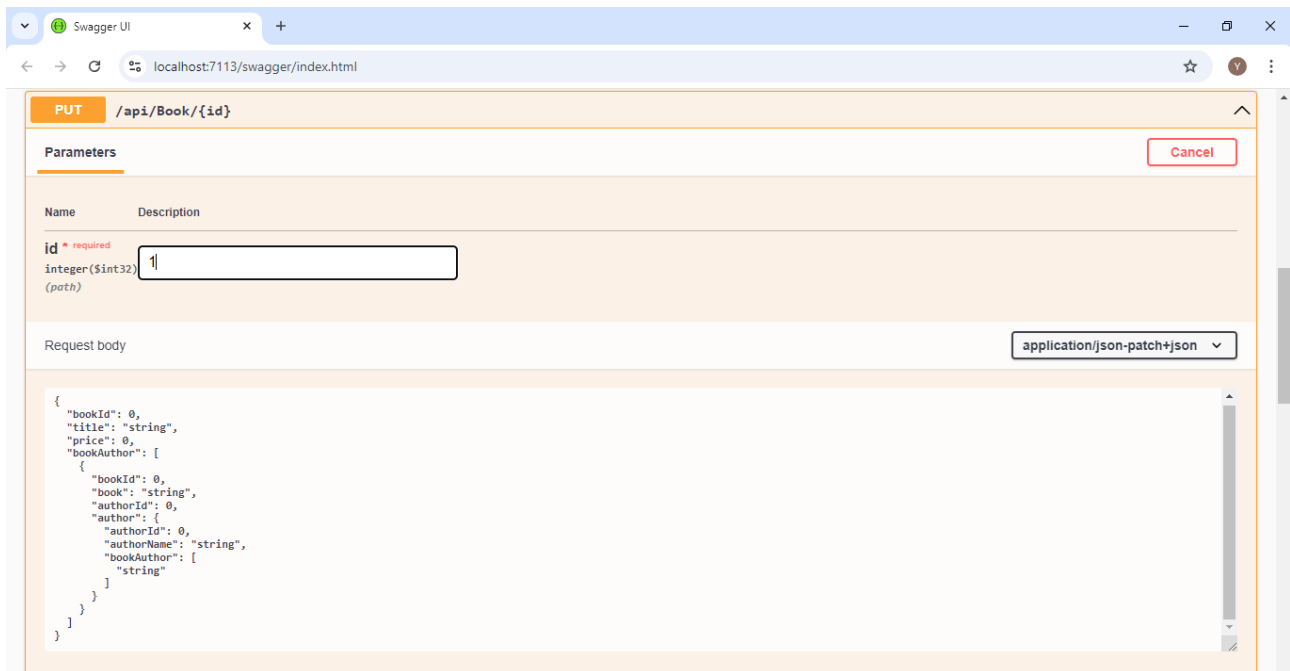
Post:



Delete:



Put:

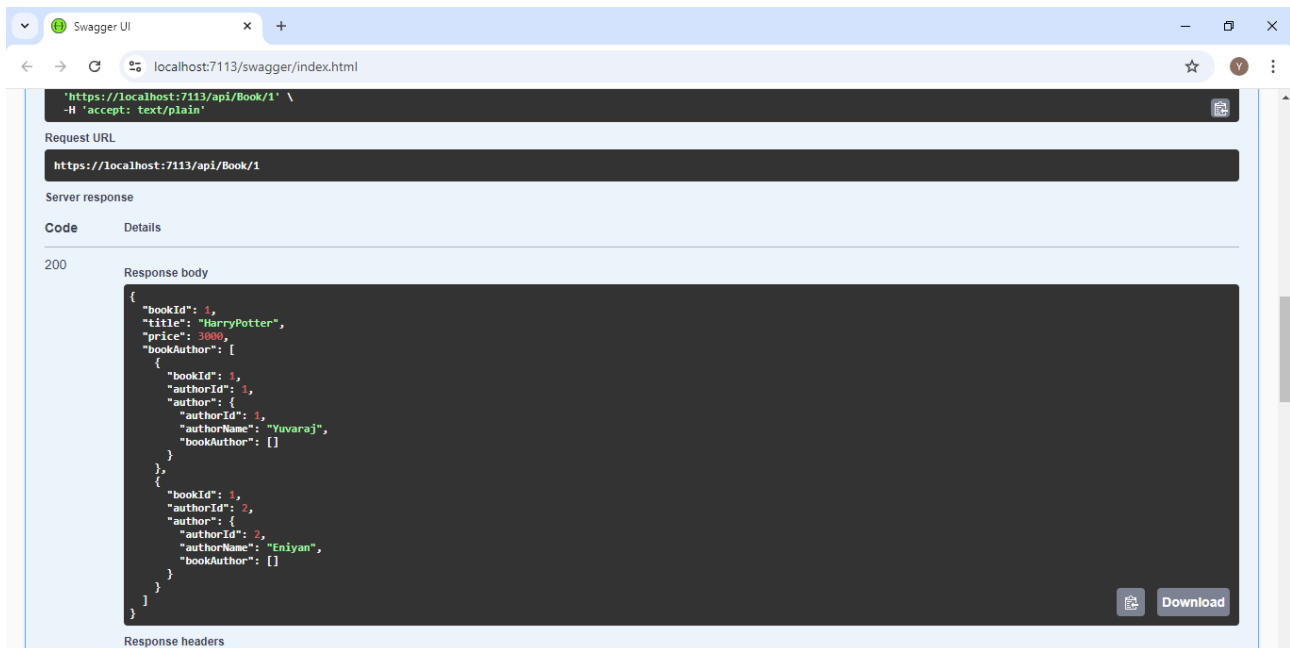


The image shows the Swagger UI interface for a PUT endpoint at `/api/Book/{id}`. The interface is divided into several sections:

- Method and Path:** A tab labeled "PUT" and the path `/api/Book/{id}`.
- Parameters:** A table with two columns: "Name" and "Description". It contains one parameter: `id` (required), type `integer($int32)`, and location `(path)`. A text input field next to it contains the value `1`. A "Cancel" button is in the top right.
- Request body:** A section with a dropdown menu set to `application/json-patch+json`. Below it is a text area containing a JSON patch document:

```
{
  "bookId": 0,
  "title": "string",
  "price": 0,
  "bookAuthor": [
    {
      "bookId": 0,
      "book": "string",
      "authorId": 0,
      "author": {
        "authorId": 0,
        "authorName": "string",
        "bookAuthor": [
          "string"
        ]
      }
    }
  ]
}
```

Get By Id:

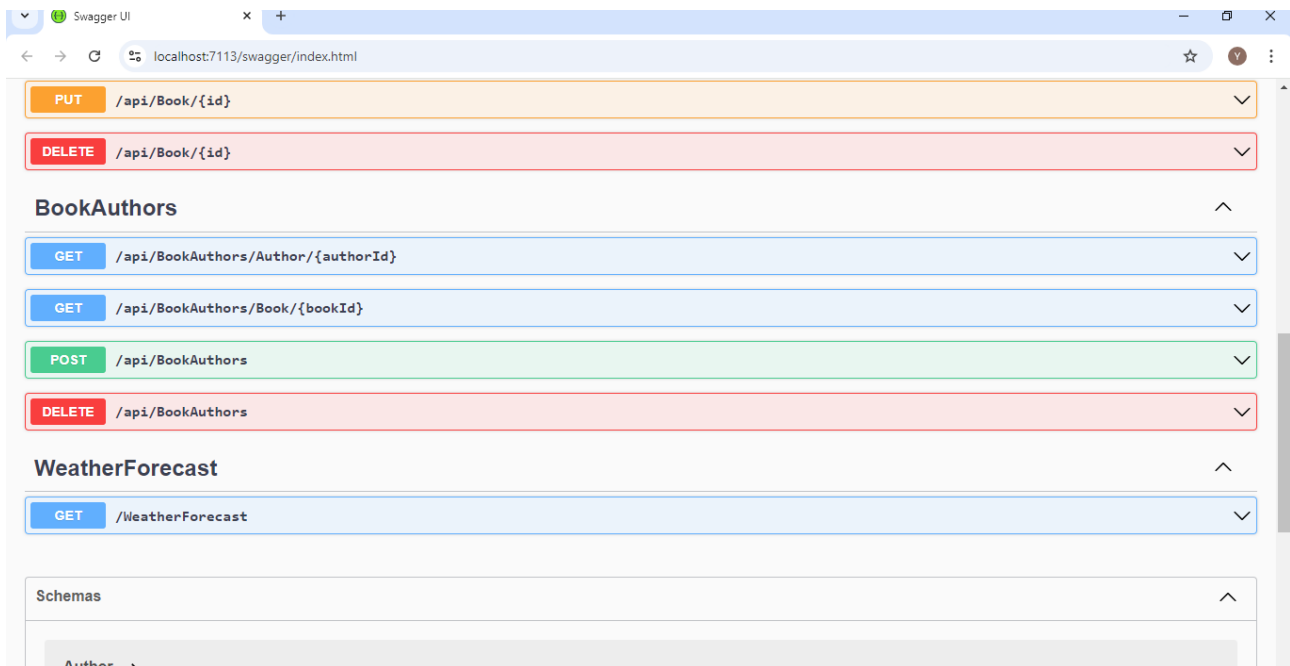


The image shows the Swagger UI interface for a GET endpoint at `/api/Book/1`. The interface displays the request and the resulting server response:

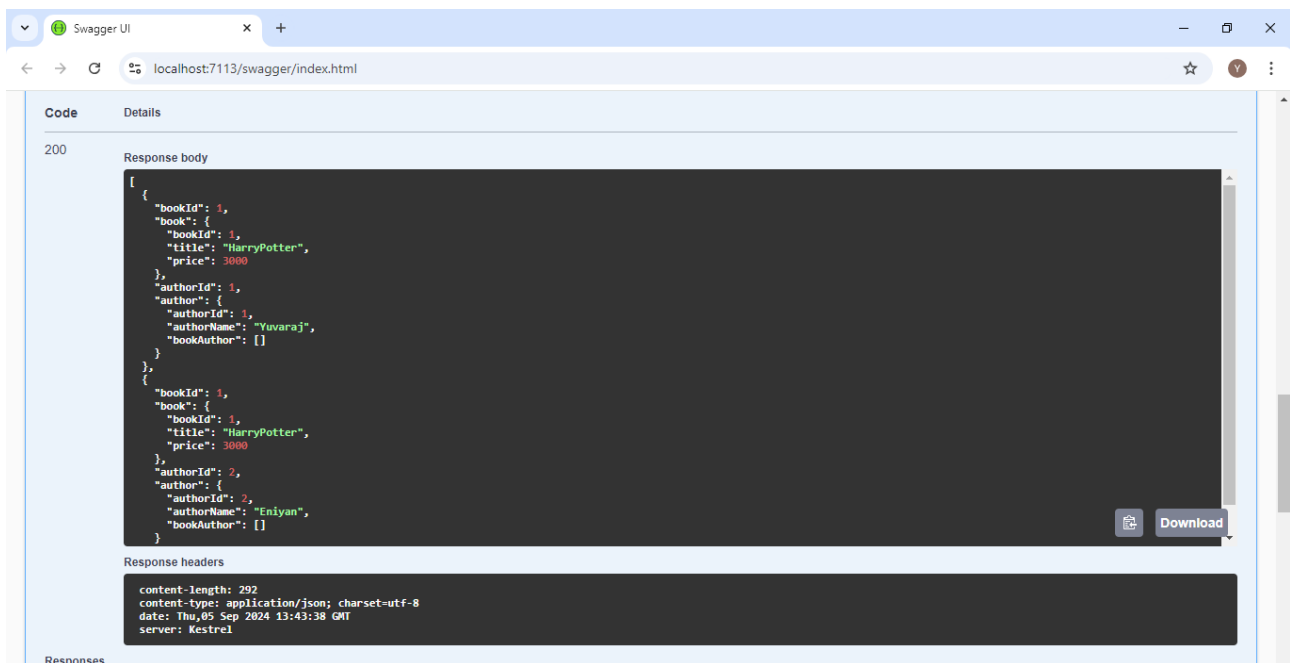
- Request URL:** `https://localhost:7113/api/Book/1`
- Server response:** A table with two columns: "Code" and "Details". It shows a `200` status code.
- Response body:** A text area containing a JSON object representing the book data:

```
{
  "bookId": 1,
  "title": "HarryPotter",
  "price": 3000,
  "bookAuthor": [
    {
      "bookId": 1,
      "authorId": 1,
      "author": {
        "authorId": 1,
        "authorName": "Yuvaraj",
        "bookAuthor": []
      }
    },
    {
      "bookId": 1,
      "authorId": 2,
      "author": {
        "authorId": 2,
        "authorName": "Eniyan",
        "bookAuthor": []
      }
    }
  ]
}
```
- Response headers:** A section at the bottom for viewing response headers.
- Buttons:** A "Download" button is located at the bottom right of the response body area.

BookAuthor Controller:



Get By BookId:



Get By AuthorId:

The screenshot shows the Swagger UI interface in a web browser. The address bar displays 'localhost:7113/swagger/index.html'. The 'Request URL' field contains 'https://localhost:7113/api/BookAuthors/Author/1'. The 'Server response' section shows a '200' status code. The 'Response body' is a JSON object representing a book and its author. The 'Response headers' section lists 'content-length: 147', 'content-type: application/json; charset=utf-8', 'date: Thu, 05 Sep 2024 13:44:20 GMT', and 'server: Kestrel'.

```
curl -X 'GET' \
  'https://localhost:7113/api/BookAuthors/Author/1' \
  -H 'accept: */*'

Request URL
https://localhost:7113/api/BookAuthors/Author/1

Server response
Code    Details
200

Response body
[
  {
    "bookId": 1,
    "book": {
      "bookId": 1,
      "title": "HarryPotter",
      "price": 3000,
      "bookAuthor": []
    },
    "authorId": 1,
    "author": {
      "authorId": 1,
      "authorName": "Yuvaraj"
    }
  }
]

Response headers
content-length: 147
content-type: application/json; charset=utf-8
date: Thu, 05 Sep 2024 13:44:20 GMT
server: Kestrel

Responses
```