

API Bookfinder

Version 0.9.1.1

Introduktion

Detta API är för närvarande i version 0.9.1.1. Det tillåter användare att interagera med en databas för en bokhandel, inklusive operationer för att hantera böcker och författare.

Det är byggt med Node.js, Express och MongoDB. Alla URL:er som refereras i dokumentationen har basen `http://localhost:3000/api`.

Denna bas-URL kommer att ändras beroende på driftsmiljön. Detta API använder ännu inga API-nycklar för åtkomst.

Felhantering:

200 OK - Begäran har lyckats.

400 Felaktig Begäran - Servern kunde inte förstå begäran på grund av ogiltig syntax.

404 Hittades Inte - Den begärda resursen kunde inte hittas.

500 Internt Serverfel - Servern stötte på ett problem som den inte vet hur den skall hantera.

Books

Beskrivning:

Returnerar alla böcker I JSON format

HTTP-metod:

GET

URL:

http://localhost:3000/api/books

Exempel på svar:

```
{
  "total": 67,
  "books": [
    {
      "_id": "66245b069b8e686f67f2fefc",
      "Name": "Awesome Steel Chair 1",
      "ISBN": 3629663257859,
      "Price": 353,
      "Genre": "Jazz",
      "ReleaseDate": "2021-08-12T00:00:00.000Z"
    },
    {
      "_id": "66245b069b8e686f67f2ff00",
      "Name": "Modern Metal Shoes",
      "ISBN": 4058170585427,
      "Price": 371,
      "Genre": "Pop",
      "ReleaseDate": "2015-07-28T23:25:29.013Z"
    },
  ],
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

Books - books by id

Beskrivning:

Ingen beskrivning tillgänglig.

HTTP-metod:

GET

URL:

<http://localhost:3000/api/books/66245b069b8e686f67f2fefc>

Exempel på förfrågan:

Exempel på svar:

```
{
  "_id": "66245b069b8e686f67f2fefc",
  "Name": "Awesome Steel Chair 1",
  "ISBN": 3629663257859,
  "Price": 353,
  "Genre": "Jazz",
  "ReleaseDate": "2021-08-12T00:00:00.000Z"
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

Books

Beskrivning:

Creates a new book entry in the database.

- **URL**: `/books`
- **Method**: `POST`
- **Required Headers**: `Content-Type: application/json`
- **Body Parameters**:
 - `Name`: The name of the book.
 - `ISBN`: The book's ISBN number.
 - `Price`: The price of the book.
 - `Genre`: The genre of the book.
 - `ReleaseDate`: The release date of the book.

HTTP-metod:

POST

URL:

<http://localhost:3000/api/books/>

Exempel på förfrågan:

```
{
  "Name": "En bok med #&&&",
  "ISBN": 123456,
  "Price": 3000,
  "Genre": "drama",
  "ReleaseDate": "2024-01-20"
}
```

Exempel på svar:

```
{
  "Name": "En bok med #&&&",
  "ISBN": 123456,
  "Price": 3000,
  "Genre": "drama",
  "ReleaseDate": "2024-01-20T00:00:00.000Z",
  "_id": "662cf4430e808f6fe35bd0f2"
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annars betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

Books - ändra books by id

Beskrivning:

Update a Book

Updates an existing book entry in the database.

- **URL**: `/books/:id``
- **Method**: `PUT``
- **Required Headers**: `Content-Type: application/json``
 - **URL Parameters**:
 - `id`_(required)_`: The ID of the book to update.
- **Body Parameters**:
 - Any book field (`Name``, `ISBN``, `Price``, `Genre``, `ReleaseDate``) that needs updating.

HTTP-metod:

PUT

URL:

`http://localhost:3000/api/books/66245b069b8e686f67f2fefc`

Exempel på förfrågan:

Kropp:

```
{
  "Name": "Awesome Steel Chair 1",
  "ISBN": 3629663257859,
  "Price": 353,
  "Genre": "Jazz",
  "ReleaseDate": "2021-08-12"
}
```

Exempel på svar:

```
{
  "updated": true,
  "book": {
    "_id": "66245b069b8e686f67f2fefc",
    "Name": "Awesome Steel Chair 1",
    "ISBN": 3629663257859,
    "Price": 353,
    "Genre": "Jazz",
    "ReleaseDate": "2021-08-12T00:00:00.000Z"
  }
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

Books - delete books by id

Beskrivning:

Delete a Book

Deletes a book entry from the database.

- **URL**: `/books/:id`
 - **Method**: `DELETE`
- **URL Parameters**:
 - `id`_(required)_: The ID of the book to delete.

HTTP-metod:

DELETE

URL:

<http://localhost:3000/api/books/66245b069b8e686f67f2ff0c>

Exempel på förfrågan:

Exempel på svar:

Vid framgångsrik utförande, skulle svaret inkludera relevanta dataobjekt eller bekräftelse av genomförda åtgärder.

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annars betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på server

authors - authors

Beskrivning:

Get all Author

- **URL**: `/authors`

- **Metod**: `GET`

- **Parametrar**: Inga

- **Exempel på begäran**:

HTTP-metod:

GET

URL:

<http://localhost:3000/api/authors>

Exempel på förfrågan:

Exempel på svar:

```
[
  {
    "_id": "66245b069b8e686f67f2fefe",
    "author": "Alfredo Koepp 1",
    "book": "Awesome Steel Chair"
  },
  {
    "_id": "66245b069b8e686f67f2ff06",
    "author": "Luke Hegmann",
    "book": "Gorgeous Plastic Gloves"
  },
  {
    "_id": "66245b069b8e686f67f2ff0a",
    "author": "Corey Gleichner",
    "book": "Refined Soft Gloves"
  }
]
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

authors - authors by id

Beskrivning:

Ingen beskrivning tillgänglig.

HTTP-metod:

GET

URL:

<http://localhost:3000/api/authors/id>

Exempel på förfrågan:

<http://localhost:3000/api/authors/661c40e1fee634aff5010872>

Exempel på svar:

```
{  
  "_id": "66245b069b8e686f67f2fefe",  
  "author": "Alfredo Koepp 1",  
  "book": "Awesome Steel Chair"  
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

authors - authors

Beskrivning:

Add an Author

Creates a new author entry in the database.

- **URL**: `/authors`
- **Method**: `POST`
- **Required Headers**: `Content-Type: application/json`
- **Body Parameters**:
 - `Name`: The name of the author.
 - `Book`: The nationality of the author.

HTTP-metod:

POST

URL:

`http://localhost:3000/api/authors/`

Exempel på förfrågan:

Kropp:

```
{
  "author": "Sara",
  "book": "skit tråkig bok2"
}
```

Exempel på svar:

```
{
  "author": "Sara",
  "book": "skit tråkig bok2",
  "_id": "662cf54e0e808f6fe35bd0f9"
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annors betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

authors - authors by id

Beskrivning:

Update an Author

Updates an existing author entry in the database.

- **URL**: `/authors/:id`
- **Method**: `PUT`
- **Required Headers**: `Content-Type: application/json`
- **URL Parameters**:
 - `id` (required): The ID of the author to update.
- **Body Parameters**:
 - Any author field (`Name`, `Nationality`) that needs updating.

HTTP-metod:

PUT

URL:

<http://localhost:3000/api/authors/66245b069b8e686f67f2fefe>

Exempel på förfrågan:

Kropp:

```
{
  "author": "Alfredo Koepp 1",
  "book": "Awesome Steel Chair"
}
```

Exempel på svar:

```
{
  "_id": "66245b069b8e686f67f2fefe",
  "author": "Alfredo Koepp 1",
  "book": "Awesome Steel Chair"
}
```

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annars betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

authors - delete authors by id

Beskrivning:

Delete an Author

Deletes an author entry from the database.

- **URL**: `/authors/:id`

- **Method**: `DELETE`

- **URL Parameters**:

- `id`_(required): The ID of the author to delete.

HTTP-metod:

DELETE

URL:

<http://localhost:3000/api/authors/66245b069b8e686f67f2ff02>

Exempel på förfrågan:

Kropp:

Exempel på svar:

Vid framgångsrik utförande, skulle svaret inkludera relevanta dataobjekt eller bekräftelse av genomförda åtgärder.

Felkoder:

200 OK - Framgång

400 Felaktig förfrågan - Förfrågan var ogiltig eller kan inte annars betjänas

404 Hittades inte - Den begärda resursen hittades inte

500 Internt serverfel - Ett generiskt fel inträffade på servern

