

Data Models, API Endpoints, Request and Response

Aman Kumar
R19112026
MCA - 19

DATA MODELS:

- I am working on Computer Vision based Project. Since, that project doesn't have any API, I will be creating basic CRUD API endpoints for custom Data Model.
- **Library Website** – I will be creating data models for library website that allows you to see the list of all available books, add new book in inventory, update a book detail, delete a book, see the list of all authors, remove an author, rent books, see the list of all books rented.

SCHEMAS:

1. Book

Property	Type	Required
name	String	True
description	String	True
price	Number	True
author	String	True
publishDate	Date	True
available	Boolean	True

2. Rented Books

Property	Type	Required
user	<pre>{ name: String, phone_no : String, street: String, city: String, postalCode: String, }</pre>	True
rentedBooks	<pre>[{ id: String, name: String, price: Number, author: String, }]</pre>	True

3. Author

Property	Type	Required
name	String	True
writing_since	Number	True
speciality_genre	String	True

REST APIs:

Request Object:

GET /available_books -> get the list of all available books

Response Object:

```
[  
  {  
    "_id": String,  
    "name": String,  
    "description": String,  
    "price": Number,  
    "author": String,  
    "publishDate": Date  
  },  
]
```

Request Object:

DELETE /book/:id -> Delete an existing book

Body:

```
{  
  "id": String,  
  "name": String,  
  "description": String,  
  "price": Number,  
  "author": String,  
  "publishDate": Date,  
}
```

Response:

Book with <id> was deleted successfully.

Request Object:

POST /book -> Add a new book

Body:

```
{  
  "name": String,  
  "description": String,  
  "price": Number,  
  "author": String,  
  "publishDate": Date,  
  "available": Boolean  
}
```

Response Object:

```
[  
  {  
    "_id": String,  
    "name": String,  
    "description": String,  
    "price": Number,  
    "author": String,  
    "publishDate": Date  
  },  
]
```

Request Object:

POST /rentbook -> rent books

Body:

```
{
  "user": {
    name: String,
    phone_no : String,
    street: String,
    city: String,
    postalCode: String,
  },
  "rentedBooks ": [{
    "author": String,
    "id": String,
    "name": String,
    "price": Number,
  }]
}
```

Response object:

```
{
  "user": {
    "name": String,
    "street": String,
    "city": String,
    "postalCode": String,
    "phone_no" : String,
    "_id": String
  },
  "rentedBooks ": [
    {
      "author": String,
      "id": String,
      "name": String,
      "price": Number,
      "_id": String
    }
  ],
  "_id": String,
  "__v": Number
}
```

Request Object:

GET /rentedBooks -> Get the list of all rented books

Response object:

```
[
  {
    "_id": String,
    "user": {
      "name": String,
      "street": String,
      "city": String,
      "postalCode": String,
      "phone_no" : String,
      "_id": String
    },
    "rentedBooks": [{
      "author": String,
      "id": String,
      "name": String,
      "price": Number,
      "_id": String
    }],
    "__v": Number
  }
]
```

Request Object:

GET /authors -> Get the list of authors

Response object:

```
[
  {
    "_id": String,
    "name": String,
    "writing_since": Number,
    "speciality_genre": String
  },
]
```

Request Object:

DELETE /authors/:id -> Delete an existing author

Body:

```
{
  "id": String,
  "name": String,
  "writing_since": Number,
  "speciality_genre": String
}
```

Response:

Author with <id> was successfully deleted.

Request Object:

POST /author -> Add a new author

Body:

```
{
  "name": String,
  "writing_since": Number,
  "speciality_genre": String
}
```

Response Object:

```
[
  {
    "_id": String,
    "name": String,
    "writing_since": Number,
    "speciality_genre": String
  },
]
```