Book Library Service API

Overview

Book Library Service APIs developed by Team Alpaca.

Version information

Version : 1.0

URI scheme

BasePath : /

Tags

- Book APIs : Book APIs
- Book List API : Book List API
- Loan APIs : Loan APIs
- Note APIs : Note APIs
- User API : User API
- User Alert API : User Alert API

Consumes

- `application/json`

Produces

- `application/json`

Paths

Returns list of books that need to be returned by input user_id and sends reminder email to that user

```
POST /alert/
```

Parameters

Type	Name	Description	Schema
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
204	No books to return	No Content
400	Syntax Error	No Content

Tags

- User Alert API

Return a list of users who need to return the loaned books

```
GET /alert/
```

Responses

HTTP Code	Description	Schema
200	Success	No Content

Tags

- User Alert API

Add a book

```
POST /book/
```

Parameters

Type	Name	Description	Schema
Query	**book_author** *optional*	book author	string
Query	**book_genre** *optional*	book genre	string
Query	**book_title** *optional*	book title	string
Query	**book_year** *optional*	book publish year	string

Responses

HTTP Code	Description	Schema
201	Book added	No Content
400	Syntax error	No Content

Tags

- Book APIs

Search book by parameters

```
GET /book/
```

Parameters

Type	Name	Description	Schema
Query	**available** *optional*	1 is available and 0 is NOT available	string
Query	**book_author** *optional*	book author	string
Query	**book_genre** *optional*	book genre	string
Query	**book_id** *optional*	book id	string
Query	**book_title** *optional*	book title	string
Query	**end_year** *optional*	Ending publish year of the book	string
Query	**start_year** *optional*	Starting publish year of the book	string

Responses

HTTP Code	Description	Schema
200	Found matching books	No Content
404	No matching books	No Content

Tags

- Book APIs

Update a book

```
PUT /book/
```

Parameters

Type	Name	Description	Schema
Query	**book_author** *optional*	book author	string
Query	**book_genre** *optional*	book genre	string
Query	**book_id** *optional*	book id	string
Query	**book_title** *optional*	book title	string
Query	**book_year** *optional*	book publish year	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
404	No such book to update	No Content

Tags

- Book APIs

Delete a book by book_id

```
DELETE /book/
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string

Responses

HTTP Code	Description	Schema
200	Deleted	No Content
400	Syntax error	No Content
404	No such book to delete	No Content

Tags

- Book APIs

Create a book list

```
POST /booklist/
```

Parameters

Type	Name	Description	Schema
Query	**description** *optional*	description of the book list	string
Query	**list_name** *optional*	name of the book list	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
201	Created	No Content
400	Syntax Error	No Content
404	No such user	No Content
409	BookList existed	No Content

Tags

- Book List API

Get the info of a book list

```
GET /booklist/
```

Parameters

Type	Name	Description	Schema
Query	**list_name** *optional*	name of the book list	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax Error	No Content
404	No such BookList	No Content

Tags

- Book List API

Delete a book list

```
DELETE /booklist/
```

Parameters

Type	Name	Description	Schema
Query	**list_name** *optional*	name of the book list	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax Error	No Content
404	No such BookList	No Content

Tags

- Book List API

Add a book to the book list

```
PUT /booklist/books
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**list_name** *optional*	name of the book list	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax Error	No Content
404	Book or BookList not found	No Content

Tags

- Book List API

Delete a book from the book list

```
DELETE /booklist/books
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**list_name** *optional*	name of the book list	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax Error	No Content
404	No such Book or BookList	No Content

Tags

- Book List API

Loan a book

```
POST /loan/
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**due** *optional*	due	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
201	Loan created	No Content
400	Syntax error	No Content
403	Forbid to borrow an unavailable book	No Content
404	No such user or book	No Content

Tags

- Loan APIs

Get the info of a loan

```
GET /loan/
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**due** *optional*	due	string
Query	**loan_id** *optional*	loan id	string
Query	**return_date** *optional*	return date	string
Query	**returned** *optional*	1 is returned and 0 is NOT returned	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
404	No matching note	No Content

Tags

- Loan APIs

Update a loan

```
PUT /loan/
```

Parameters

Type	Name	Description	Schema
Query	**due** *optional*	due	string
Query	**loan_id** *optional*	loan id	string
Query	**return_date** *optional*	return date	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
404	No such loan or book	No Content

Tags

- Loan APIs

Delete a loan by loan_id

```
DELETE /loan/
```

Parameters

Type	Name	Description	Schema
Query	**loan_id** *optional*	loan id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
403	Forbid to delete not returned loan	No Content
404	No such loan	No Content

Tags

- Loan APIs

Add a note

```
POST /note/
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**content** *optional*	Content	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
201	Note added	No Content
400	Syntax error	No Content
404	Not found user or book	No Content

Tags

- Note APIs

Search note by parameters

```
GET /note/
```

Parameters

Type	Name	Description	Schema
Query	**book_id** *optional*	book id	string
Query	**content** *optional*	content	string
Query	**note_id** *optional*	note id	string
Query	**user_id** *optional*	user id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
404	No matching notes	No Content

Tags

- Note APIs

Update a content of a note

```
PUT /note/
```

Parameters

Type	Name	Description	Schema
Query	**content** *optional*	Content	string
Query	**note_id** *optional*	note id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
404	No matching note	No Content

Tags

- Note APIs

Delete a note by note_id

```
DELETE /note/
```

Parameters

Type	Name	Description	Schema
Query	**note_id** *optional*	note id	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
404	No such note to delete	No Content

Tags

- Note APIs

Sign up

```
POST /user/
```

Parameters

Type	Name	Description	Schema
Query	**email** *optional*	email address	string
Query	**password** *optional*	password	string
Query	**user_name** *optional*	user name	string

Responses

HTTP Code	Description	Schema
201	User created	No Content
400	Syntax error	No Content
409	User name existed	No Content

Tags

- User API

Sign in

```
GET /user/
```

Parameters

Type	Name	Description	Schema
Query	**password** *optional*	password	string
Query	**user_name** *optional*	user name	string

Responses

HTTP Code	Description	Schema
200	Success	No Content
400	Syntax error	No Content
401	Unauthorized access	No Content
404	User not found	No Content

Tags

- User API