

MyLibrary API Documentation

Version: 1.0.0

Base URL: <https://api.mylibrary.com/v1>

Overview

MyLibrary API provides access to a comprehensive digital library of books, authors, and user interactions. This RESTful API enables developers to search for books, retrieve detailed information on authors, manage user reading lists, and submit reviews.

Key Features

- Books: Retrieve book details, search books by genre, author, and title.
 - Authors: Access detailed information about authors and their works.
 - User Library: Manage user reading lists, add/remove books, and track reading progress.
 - Reviews: Post, edit, and delete reviews for any book in the library.
-

Authentication

The MyLibrary API uses API Key authentication. Pass the API key in the request header as follows:

http

Copy code

Authorization: Bearer YOUR_API_KEY

To obtain an API key, register at MyLibrary Developer Portal.

Rate Limits

Plan	Limit
Free Tier	100 requests per hour
Pro Tier	1,000 requests per hour
Enterprise Tier	10,000 requests per hour

Endpoints

Books

1. Get Book Details

Retrieve details of a specific book by its ID.

- URL: `/books/{book_id}`
- Method: `GET`
- Authentication: Required
- Parameters:
 - Path Parameter:
 - `book_id` (string): Unique identifier for the book.

Example Request:

http

Copy code

```
GET /v1/books/12345 HTTP/1.1
```

```
Host: api.mylibrary.com
```

```
Authorization: Bearer YOUR_API_KEY
```

•

Response:

json

Copy code

```
{
  "id": "12345",
  "title": "Pride and Prejudice",
  "author": {
    "id": "67890",
    "name": "Jane Austen"
  },
  "genre": "Classic",
  "published_year": 1813,
  "summary": "A romantic novel that critiques the British landed gentry of the early 19th century."
}
```

•

- Response Codes:
 - `200 OK` - Successful request.
 - `404 Not Found` - Book not found.
 - `401 Unauthorized` - Invalid or missing API key.

2. Search Books

Search for books based on various query parameters.

- URL: </books/search>
- Method: [GET](#)
- Authentication: Required
- Parameters:
 - Query Parameters:
 - [title](#) (string, optional): Title of the book.
 - [author](#) (string, optional): Name of the author.
 - [genre](#) (string, optional): Genre of the book.
 - [published_year](#) (int, optional): Publication year.

Example Request:

http

Copy code

[GET /v1/books/search?title=Pride HTTP/1.1](#)

[Host: api.mylibrary.com](#)

[Authorization: Bearer YOUR_API_KEY](#)

•

Response:

json

Copy code

```
{
  "total_results": 1,
  "books": [
    {
      "id": "12345",
      "title": "Pride and Prejudice",
      "author": {
        "id": "67890",
        "name": "Jane Austen"
      },
      "genre": "Classic",
      "published_year": 1813
    }
  ]
}
```

•

- Response Codes:
 - [200 OK](#) - Successful request.
 - [400 Bad Request](#) - Invalid query parameters.

Authors

Get Author Details

Retrieve details about an author.

- URL: `/authors/{author_id}`
- Method: `GET`
- Authentication: Required
- Parameters:
 - Path Parameter:
 - `author_id` (string): Unique identifier for the author.

Example Request:

http

Copy code

`GET /v1/authors/67890 HTTP/1.1`

`Host: api.mylibrary.com`

`Authorization: Bearer YOUR_API_KEY`

•

Response:

json

Copy code

```
{
  "id": "67890",
  "name": "Jane Austen",
  "birth_year": 1775,
  "death_year": 1817,
  "biography": "Jane Austen was an English novelist known primarily for her six major novels...",
  "books": [
    {
      "id": "12345",
      "title": "Pride and Prejudice"
    },
    {
      "id": "23456",
      "title": "Sense and Sensibility"
    }
  ]
}
```

•

- Response Codes:
 - `200 OK` - Successful request.
 - `404 Not Found` - Author not found.

User Library

Add Book to Reading List

Add a specific book to the user's reading list.

- URL: `/users/{user_id}/library`

- Method: **POST**
- Authentication: Required
- Parameters:
 - Path Parameter:
 - **user_id** (string): Unique identifier for the user.
 - Body Parameters:
 - **book_id** (string): ID of the book to be added.

Example Request:

http

Copy code

POST /v1/users/100/library HTTP/1.1

Host: api.mylibrary.com

Authorization: Bearer YOUR_API_KEY

Content-Type: application/json

```
{
  "book_id": "12345"
}
```

•

Response:

json

Copy code

```
{
  "message": "Book added to your library",
  "book_id": "12345"
}
```

•

- Response Codes:
 - **201 Created** - Book successfully added.
 - **404 Not Found** - User or book not found.

Reviews

Post a Review

Submit a review for a book.

- URL: **/books/{book_id}/reviews**
- Method: **POST**
- Authentication: Required
- Parameters:
 - Path Parameter:
 - **book_id** (string): ID of the book being reviewed.
 - Body Parameters:
 - **user_id** (string): ID of the user posting the review.

- **rating** (int, required): Rating out of 5.
- **review** (string, optional): Text of the review.

Example Request:

http

Copy code

```
POST /v1/books/12345/reviews HTTP/1.1
```

```
Host: api.mylibrary.com
```

```
Authorization: Bearer YOUR_API_KEY
```

```
Content-Type: application/json
```

```
{
  "user_id": "100",
  "rating": 5,
  "review": "An amazing classic that still holds relevance today."
}
```

•

Response:

json

Copy code

```
{
  "message": "Review submitted successfully",
  "review_id": "54321"
}
```

•

- Response Codes:
 - **201 Created** - Review successfully posted.
 - **400 Bad Request** - Invalid rating value.

Error Codes and Responses

Code	Description
200	OK - Request was successful
201	Created - Resource was successfully created
400	Bad Request - Invalid input parameters
401	Unauthorized - API key missing or invalid
403	Forbidden - Action not allowed for user
404	Not Found - Resource not found

Example Code

Here's a sample script using Python and `requests` to access the API:

```
python
Copy code
import requests

api_url = "https://api.mylibrary.com/v1/books/12345"
headers = {
    "Authorization": "Bearer YOUR_API_KEY"
}

response = requests.get(api_url, headers=headers)
if response.status_code == 200:
    book = response.json()
    print("Title:", book["title"])
    print("Author:", book["author"]["name"])
else:
    print("Failed to retrieve book details:", response.status_code)
```

Changelog

Version 1.0.0

- Initial release with book, author, user library, and review endpoints.
-

Support

For assistance or questions, contact MyLibrary Support at support@mylibrary.com.
