

Library Management RESTful API Documentation

1. Introduction

This is a Flask-based RESTful API for managing a Library System. It supports CRUD operations for books using MySQL as the backend database. The API allows you to create, retrieve, update, and delete books.

2. Technology Stack

- Backend: Python (Flask)
- Database: MySQL
- Tools: VS Code, Postman
- Libraries: Flask, SQLAlchemy, PyMySQL

3. Setup Instructions

1. Clone or unzip the project.
2. Navigate to the project folder.
3. Create a virtual environment (optional): `python -m venv venv`
4. Install dependencies: `pip install -r requirements.txt`
5. Run the app: `python app.py`
6. Use Postman to test the endpoints.

4. Database Schema

Database Name: librarydb

Table: books

Column Name	Data Type	Description
id	Integer	Primary Key
title	String	Book Title
author	String	Author Name

Column Name	Data Type	Description
year	Integer	Year of Release

4. Database Configuration

Update your `config.py` with the correct MySQL connection string:

Example: `mysql+pymysql://username:password@localhost/librarydb`

5. API Endpoints

POST /books

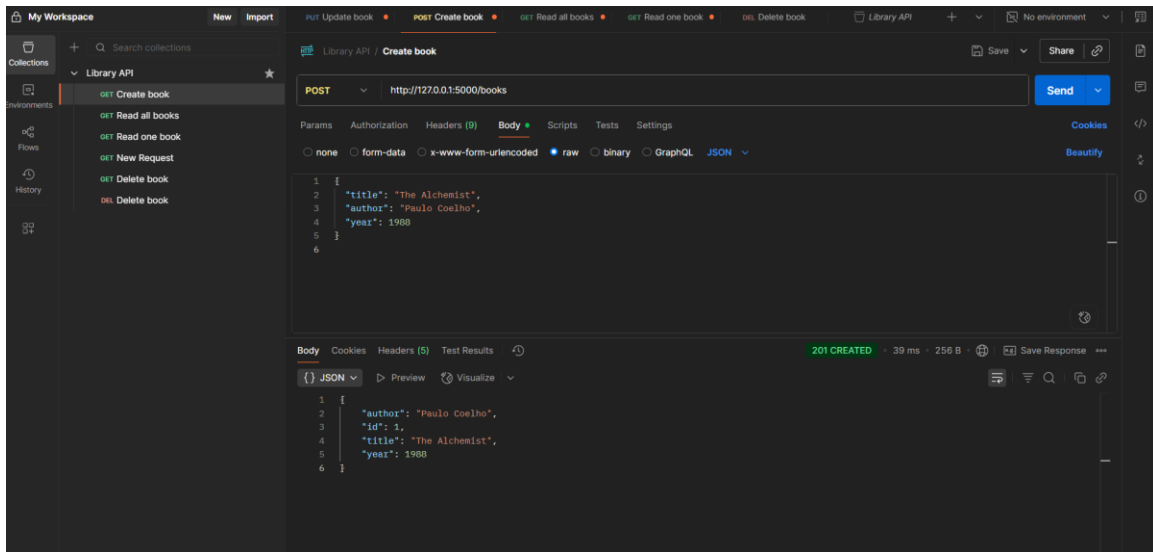
Description: Add a new book

Request JSON:

```
{
  "title": "The Alchemist",
  "author": "Paulo Coelho",
  "year": 1988
}
```

Response JSON:

```
{
  "id": 1,
  "title": "The Alchemist",
  "author": "Paulo Coelho",
  "year": 1988
}
```

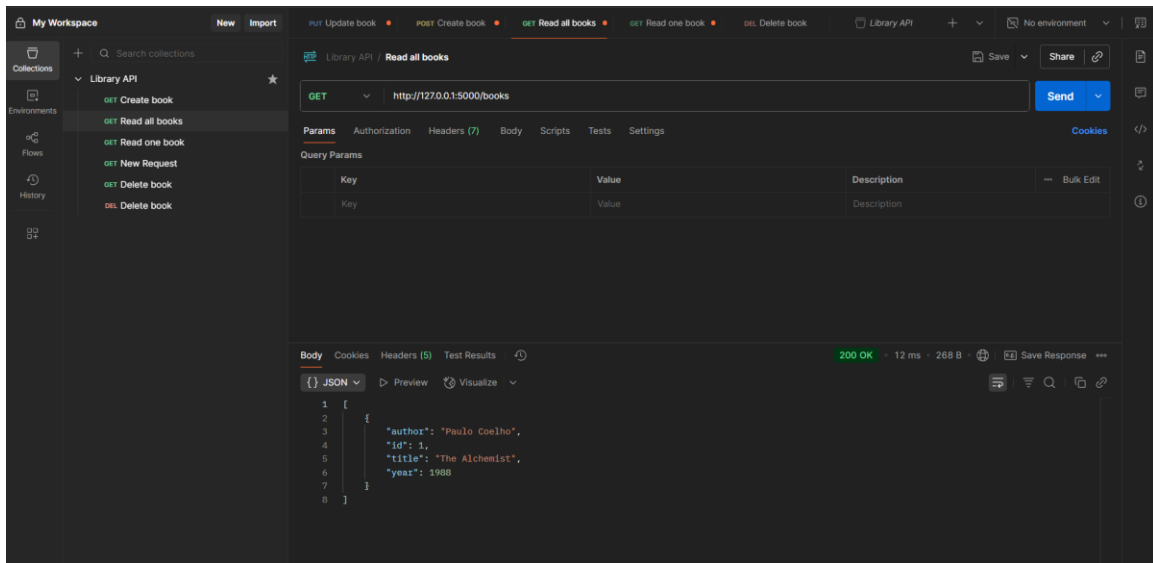


GET /books

Description: Retrieve all books

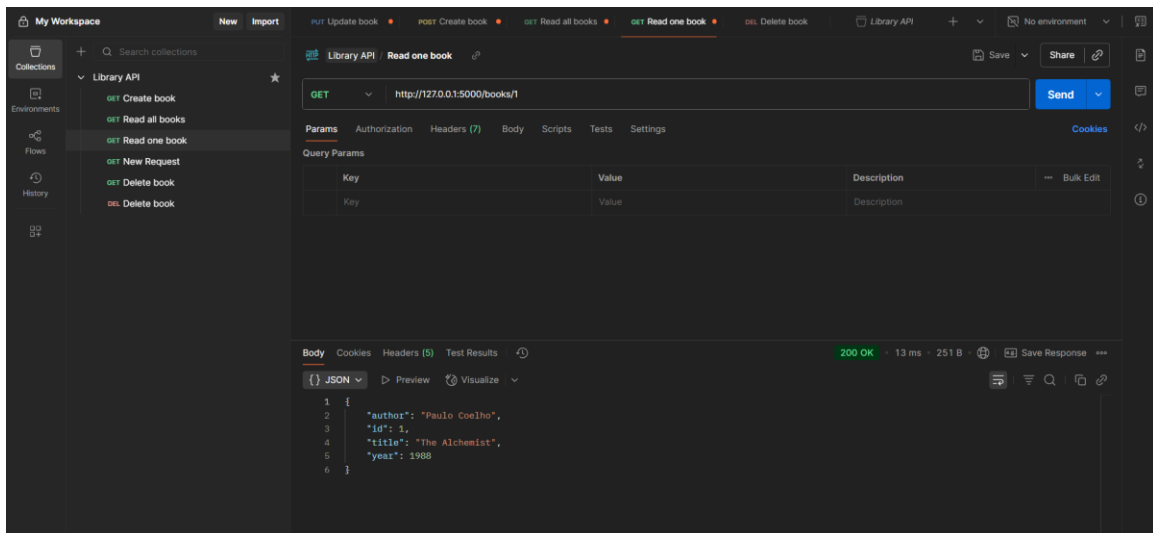
Response JSON:

```
[
  {
    "id": 1,
    "title": "The Alchemist",
    "author": "Paulo Coelho",
    "year": 1988
  }
]
```



GET /books/<id>

Description: Retrieve a book by ID

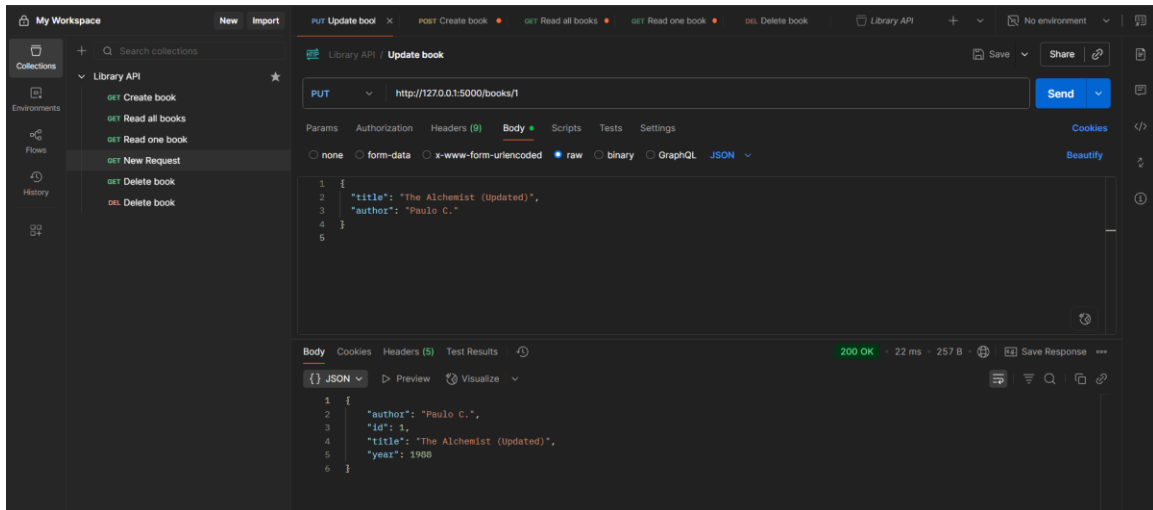


PUT /books/<id>

Description: Update an existing book

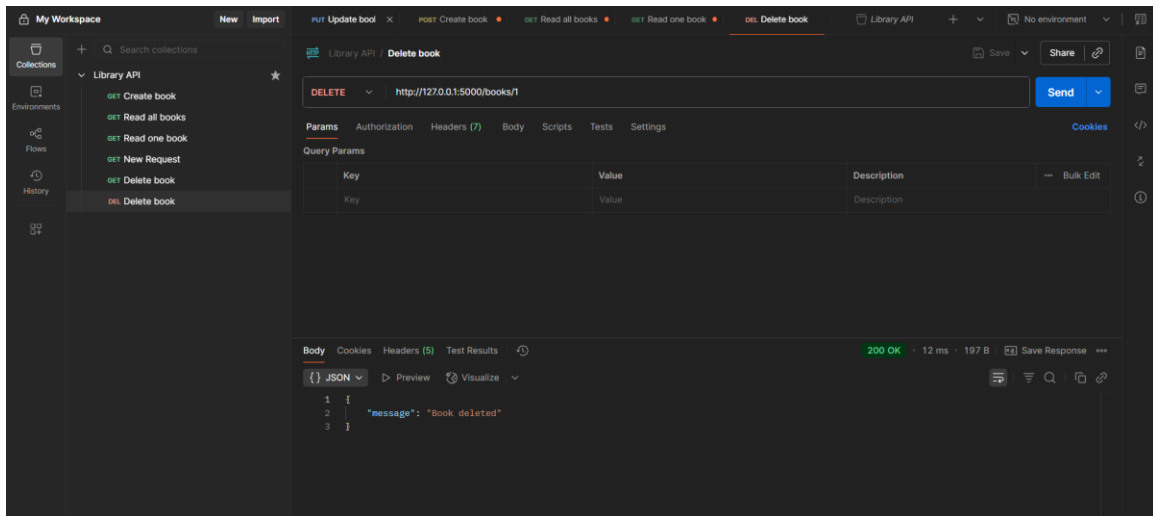
Request JSON:

```
{
  "title": "Updated Title",
  "author": "Updated Author",
  "year": 2024
}
```



DELETE /books/<id>

Description: Delete a book by ID



Prepared by Vishakha Nale
