Project code

from flask import Flask, jsonify, request

app = Flask(\_\_name\_\_)

# Create some sample data

books = [

{"id": 1, "title": "Book 1", "author": "Author 1"},

{"id": 2, "title": "Book 2", "author": "Author 2"},

{"id": 3, "title": "Book 3", "author": "Author 3"}

]

# GET /books - Get all books

@app.route('/books', methods=['GET'])

def get\_books():

return jsonify(books)

# GET /books/{id} - Get a specific book

@app.route('/books/<int:book\_id>', methods=['GET'])

def get\_book(book\_id):

book = next((book for book in books if book['id'] == book\_id), None)

if book:

return jsonify(book)

return jsonify({'message': 'Book not found'}), 404

# POST /books - Add a new book

@app.route('/books', methods=['POST'])

def add\_book():

data = request.get\_json()

new\_book = {

'id': len(books) + 1,

'title': data['title'],

'author': data['author']

}

books.append(new\_book)

return jsonify(new\_book), 201

# PUT /books/{id} - Update a book

@app.route('/books/<int:book\_id>', methods=['PUT'])

def update\_book(book\_id):

data = request.get\_json()

book = next((book for book in books if book['id'] == book\_id), None)

if book:

book['title'] = data['title']

book['author'] = data['author']

return jsonify(book)

return jsonify({'message': 'Book not found'}), 404

# DELETE /books/{id} - Delete a book

@app.route('/books/<int:book\_id>', methods=['DELETE'])

def delete\_book(book\_id):

global books

books = [book for book in books if book['id'] != book\_id]

return jsonify({'message': 'Book deleted'})

# Run the application

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)