Лабораторна робота №4 Частина 2 (№6)

З дисципліни Бази даних та інформаційні системи

Студента групи МІТ-31 Власенко Богдана

Тема: Реалізація API за допомогою Express.js та Node.js для доступу та управління даними (CRUD операції)

Метою даної лабораторної роботи було створення API з використанням Express.js та Node.js для забезпечення можливості доступу та управління даними, включаючи CRUD операції (створення, читання, оновлення, видалення). Для досягнення цієї мети ми використовували популярні інструменти та технології, такі як Express.js для створення веб-сервера та маршрутизації, та Node.js як середовище виконання.

Інформаційна система для бібліотеки.

Функціональні вимоги до такої ІС:

1. Керування книгами:

* Додавання нових книг у систему.
* Редагування інформації про книги.
* Видалення книг з системи.

1. Керування авторами:

* Додавання нових авторів у систему.
* Редагування інформації про авторів.
* Видалення авторів з системи.

1. Керування користувачами:

* Реєстрація нових користувачів у системі.
* Аутентифікація користувачів.
* Редагування інформації про користувачів.
* Видалення користувачів з системи.

1. Керування примірниками книг:

* Додавання нових примірників книг до системи.
* Оновлення інформації про примірники книг.
* Видалення примірників книг з системи.

1. Керування видачею та поверненням книг:

* Видача книг користувачам.
* Прийом та обробка повернених книг.
* Оновлення інформації про терміни видачі та повернення книг.

1. Пошук та фільтрація книг:

* Пошук книг за назвою, автором, жанром, рейтингом та роком публікації.
* Фільтрація книг за різними критеріями.

1. Статистика та звіти:

* Виведення статистичної інформації про книги, авторів, користувачів та інше.
* Формування звітів про діяльність бібліотеки.

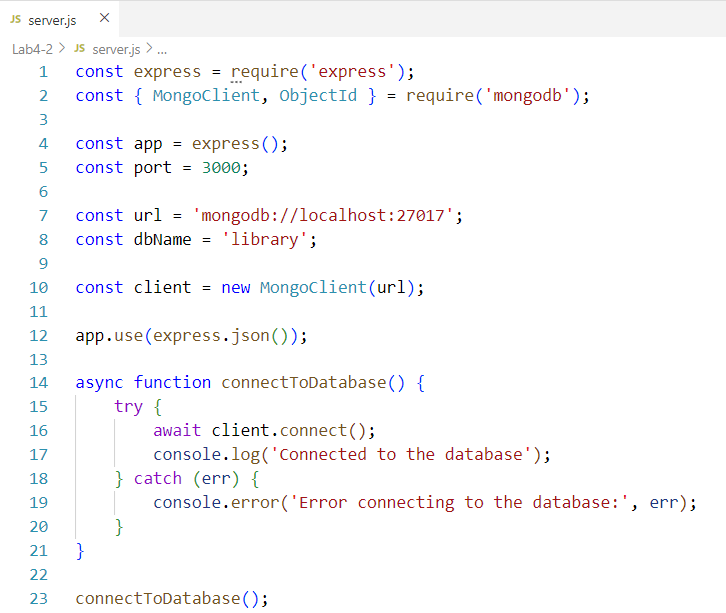
1. Система управління доступом:

* Налаштування рівнів доступу до функціональності системи для різних типів користувачів.

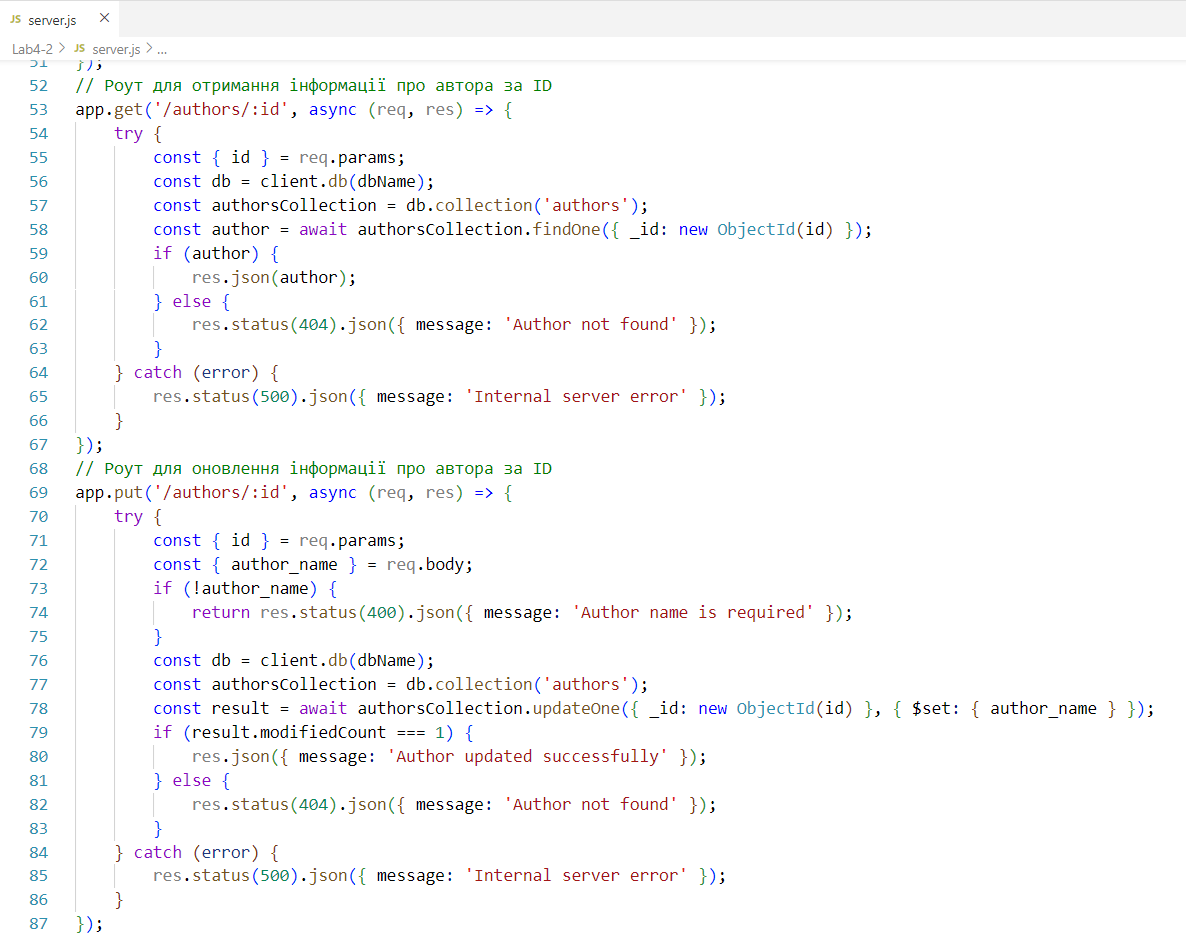
Виконання лабораторної роботи:

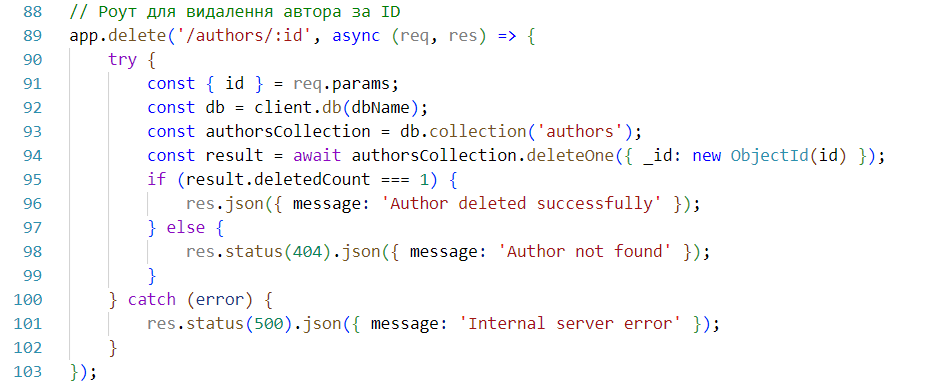
**Створення сервера і роутів**

Використовуючи Express.js, я створив веб-сервер, який слухає певний порт і обробляє HTTP запити. Я визначив маршрути, які відповідають на HTTP запити та виконують відповідні CRUD операції.





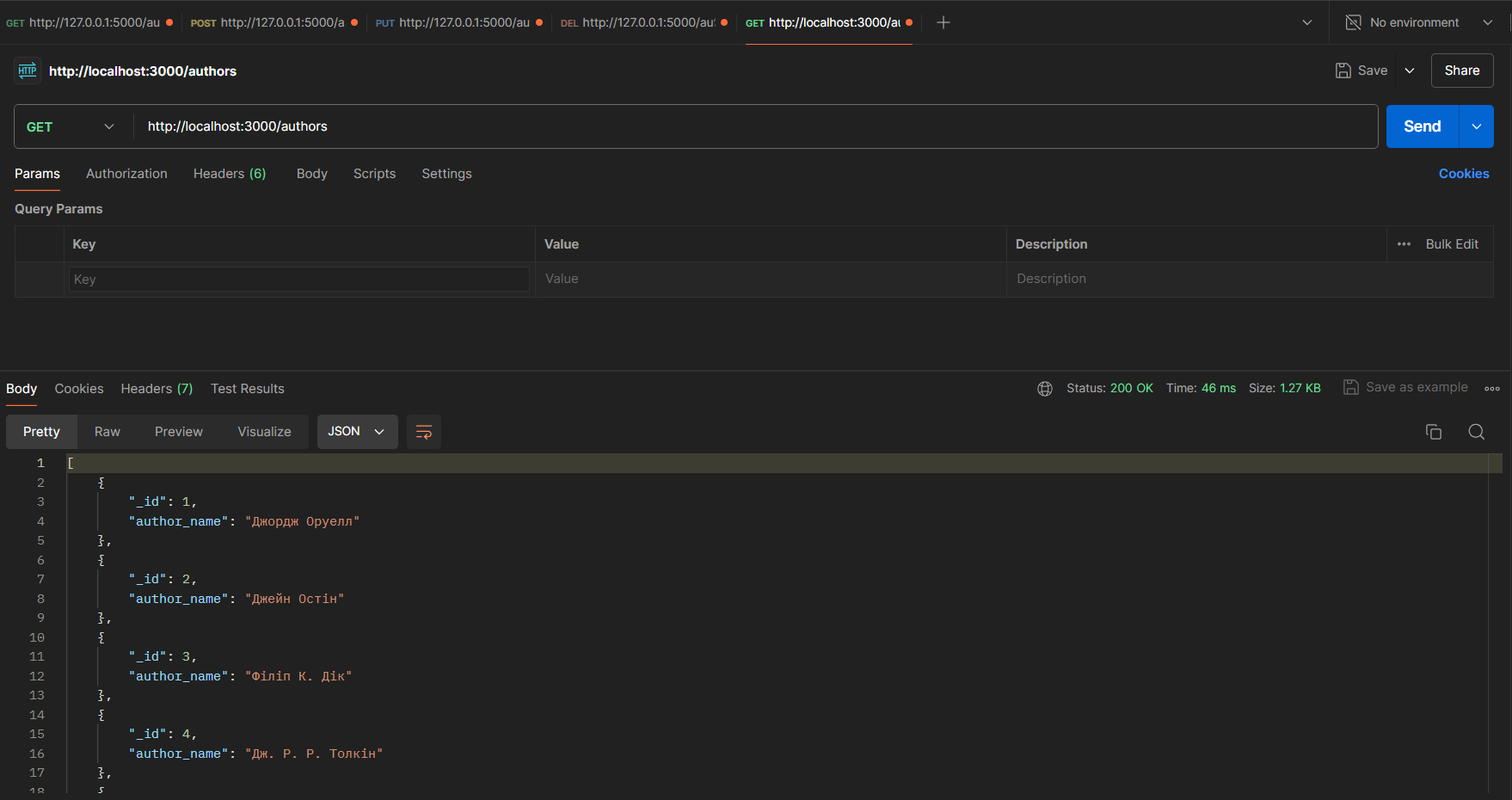


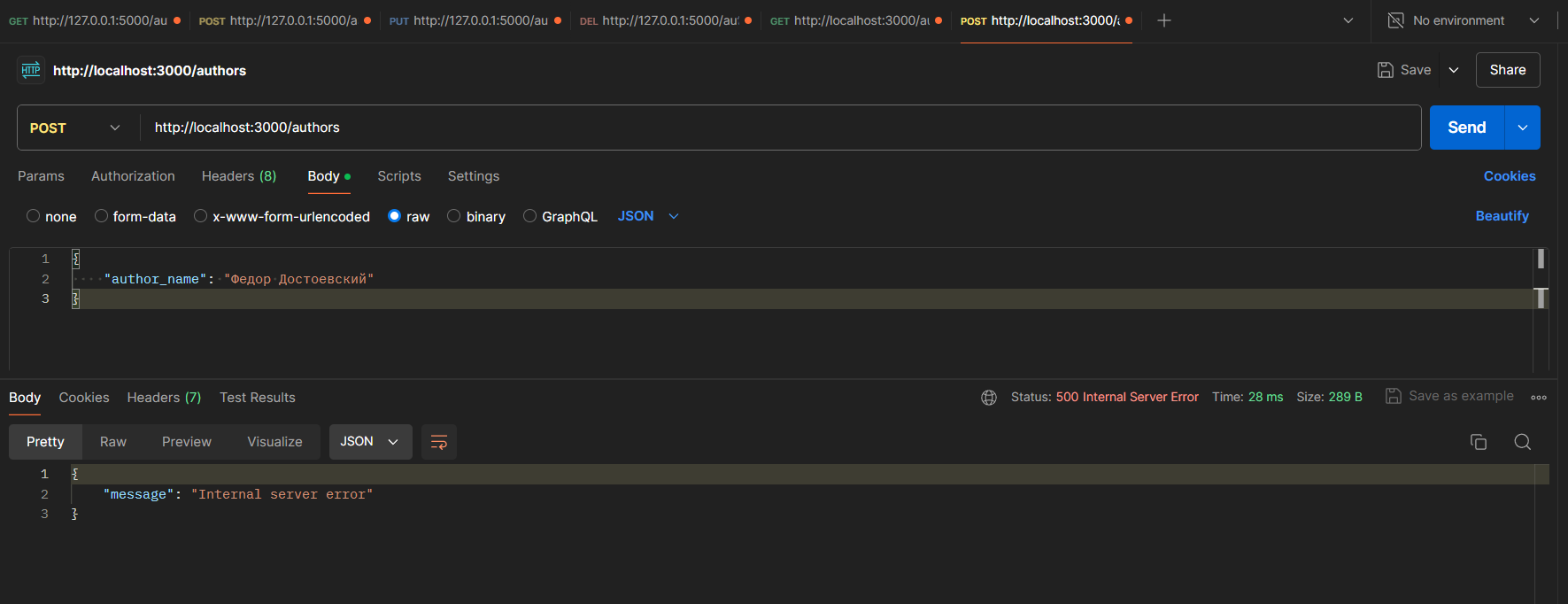


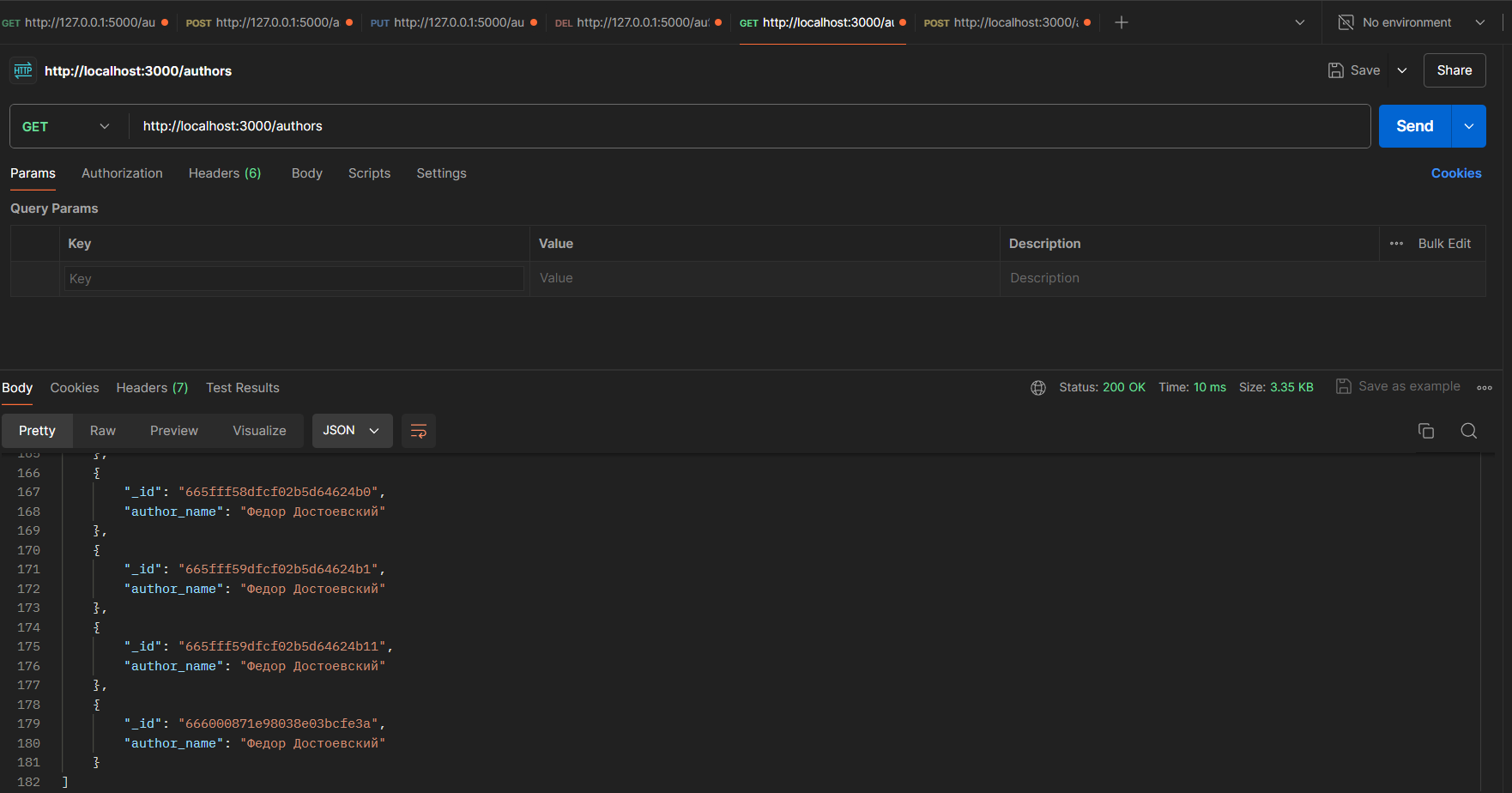
**Тестування**

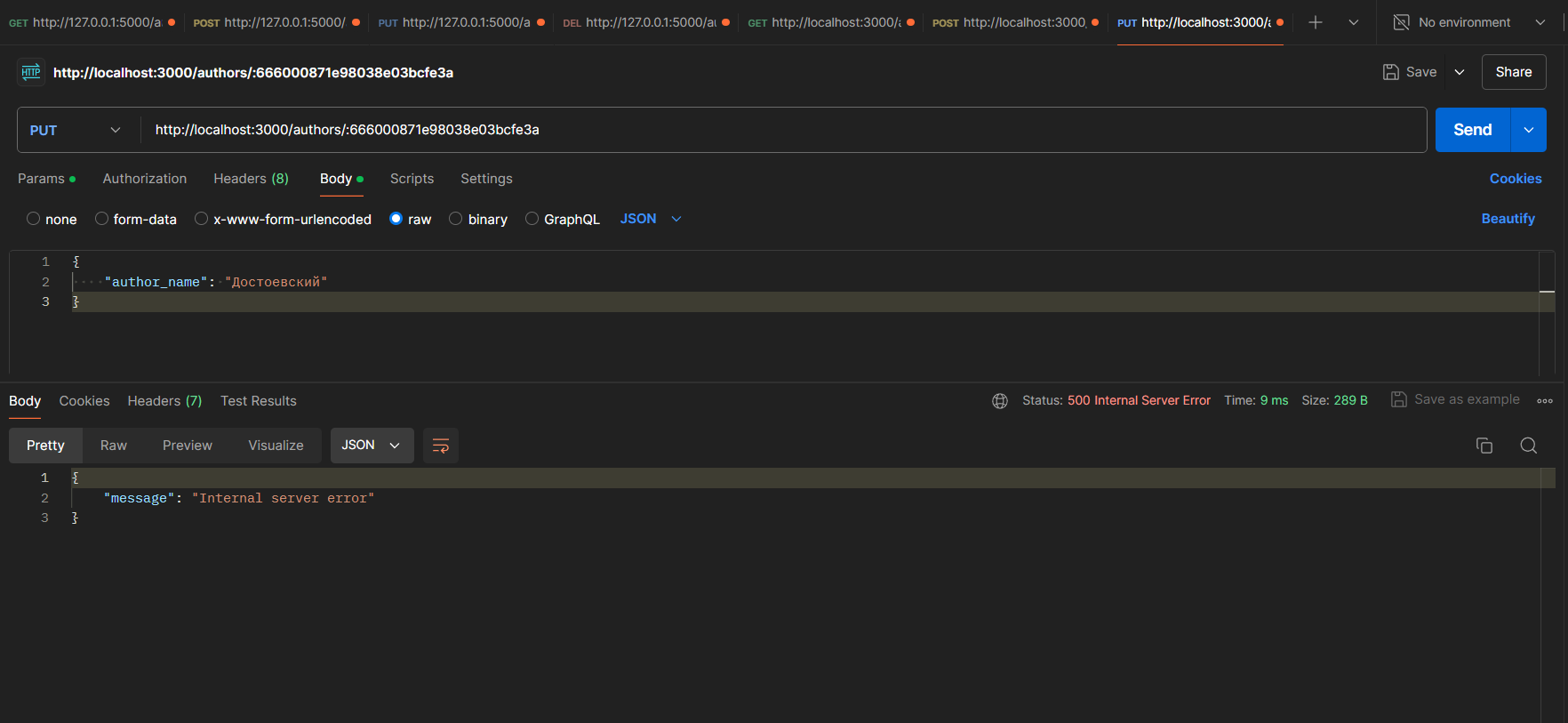
За допомогою Postman я перевірив роботу ендпоінтів, виконуючи різні HTTP запити.

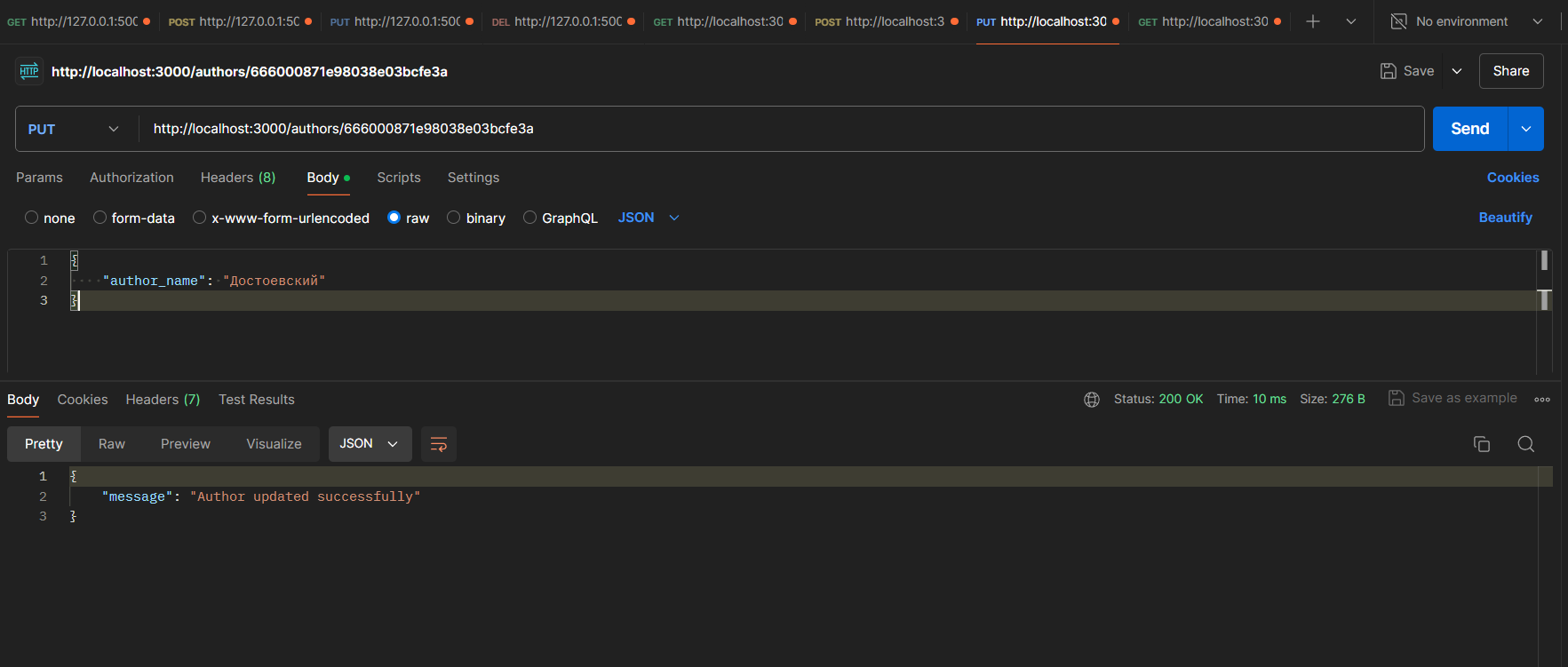


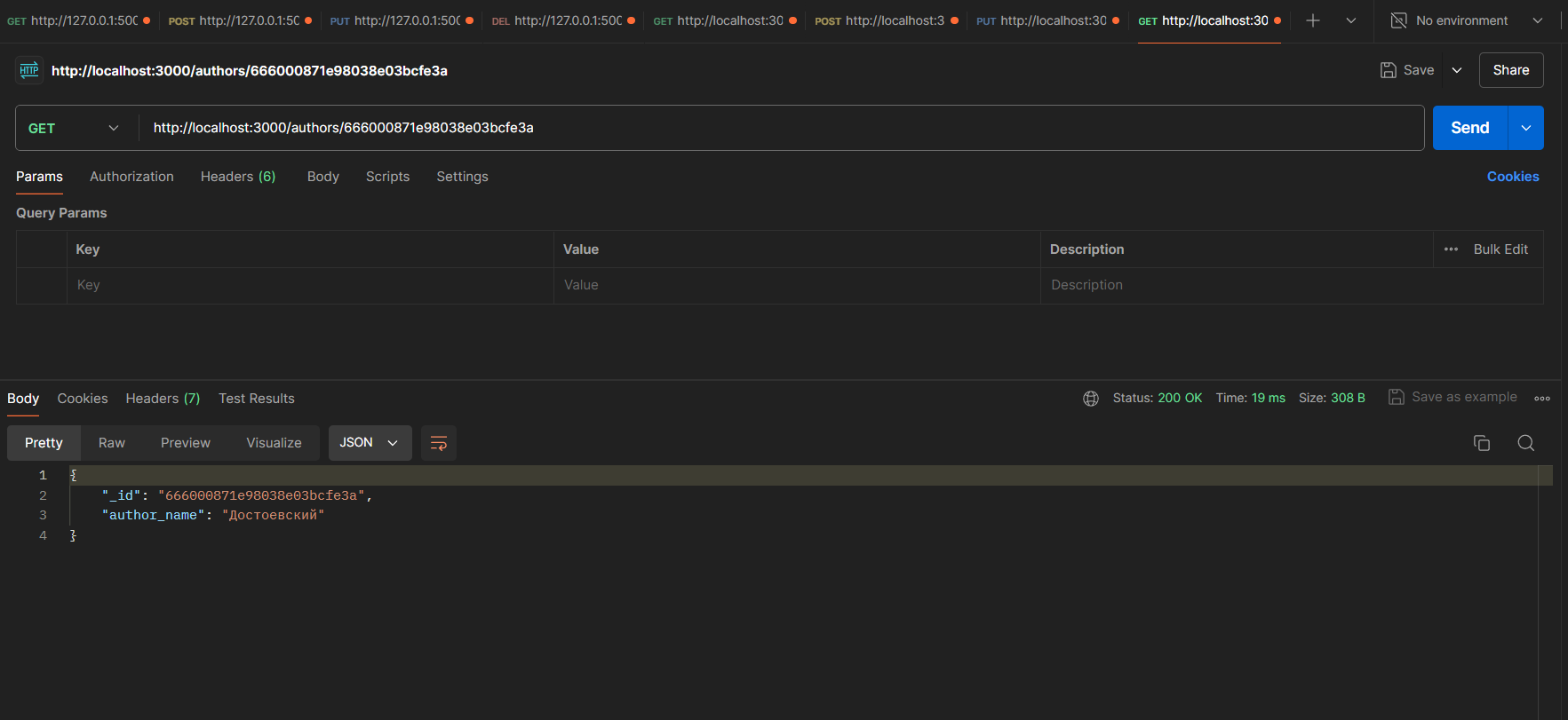


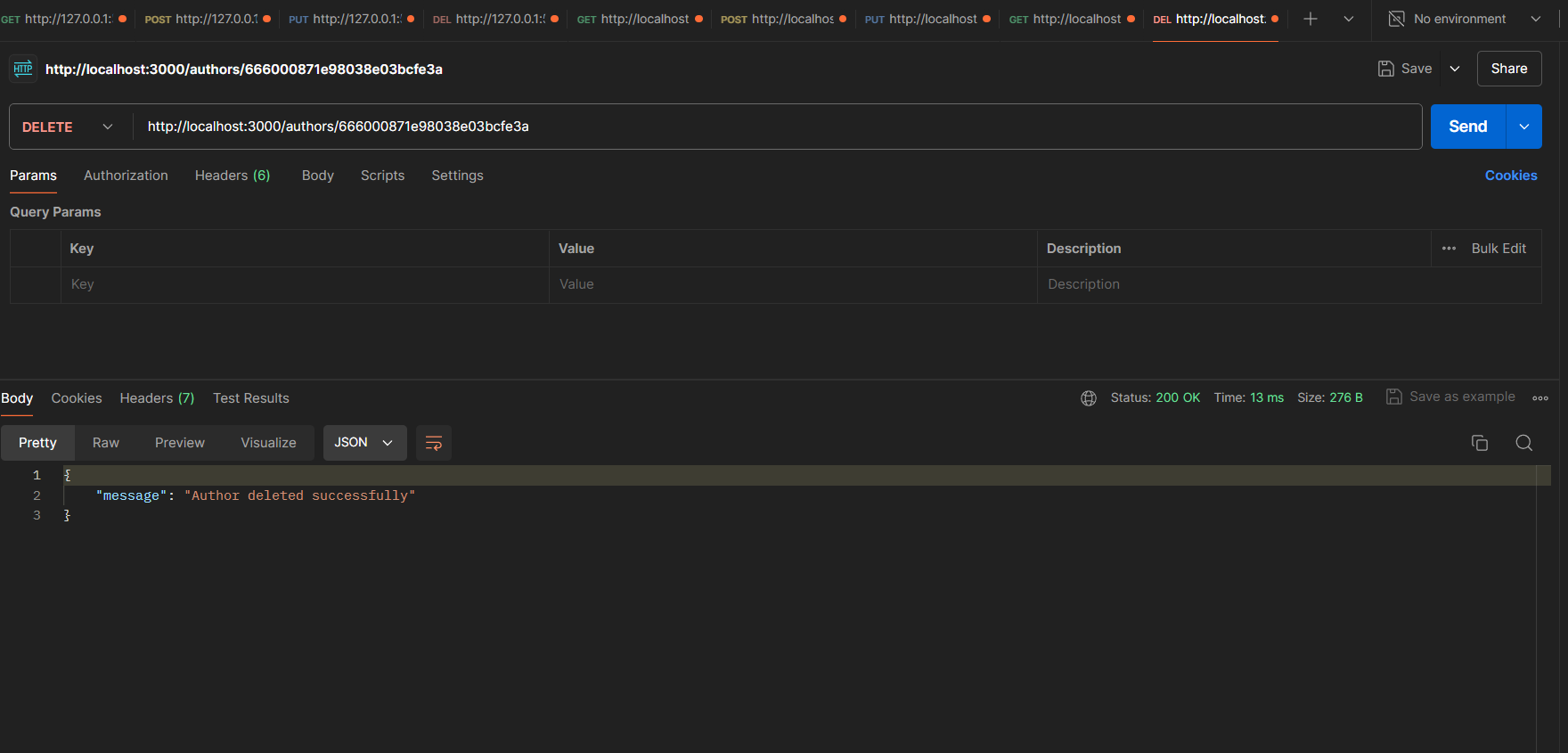


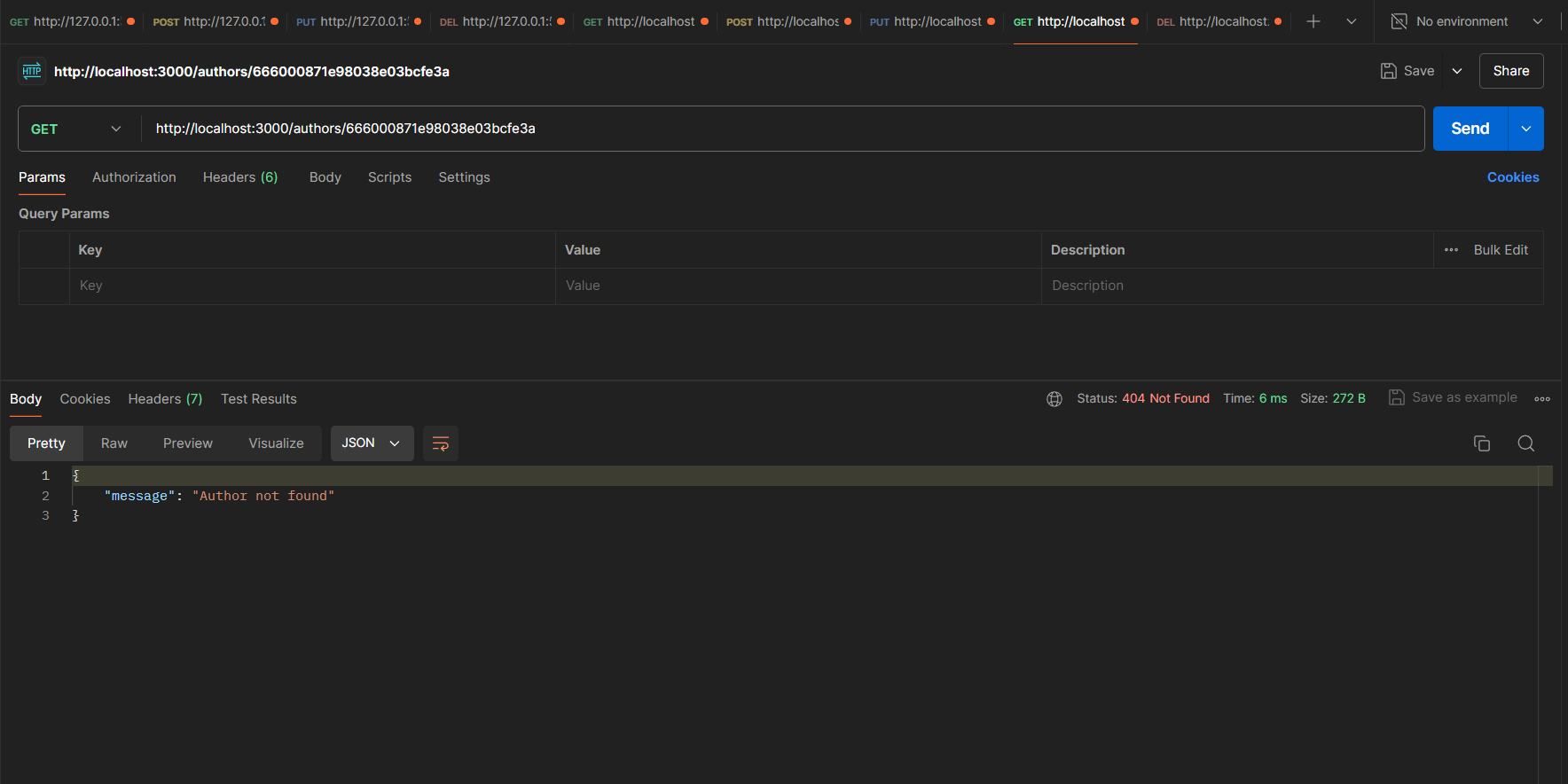












Висновки:

У результаті цієї лабораторної роботи я успішно реалізував API з використанням Express.js та Node.js для доступу та управління даними, включаючи всі CRUD операції. Я отримав поглиблене розуміння веб-розробки на базі Node.js та вміння створювати потужні API для взаємодії з даними.

Додатки:

1. setupLibraryDB.js - код для створення таблиць бази даних

2. Requests.txt – 20 запитів до бази даних

3. server.js – код API