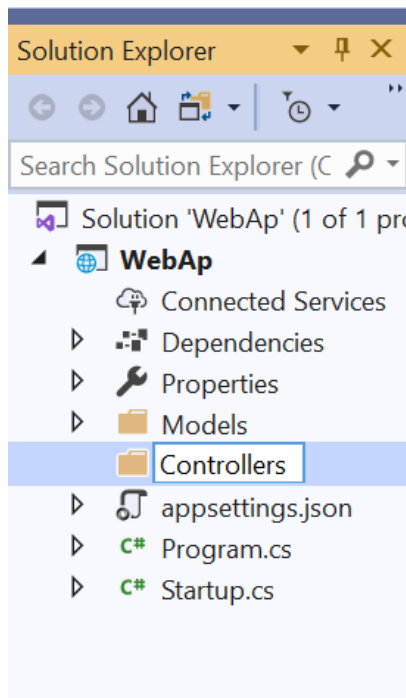
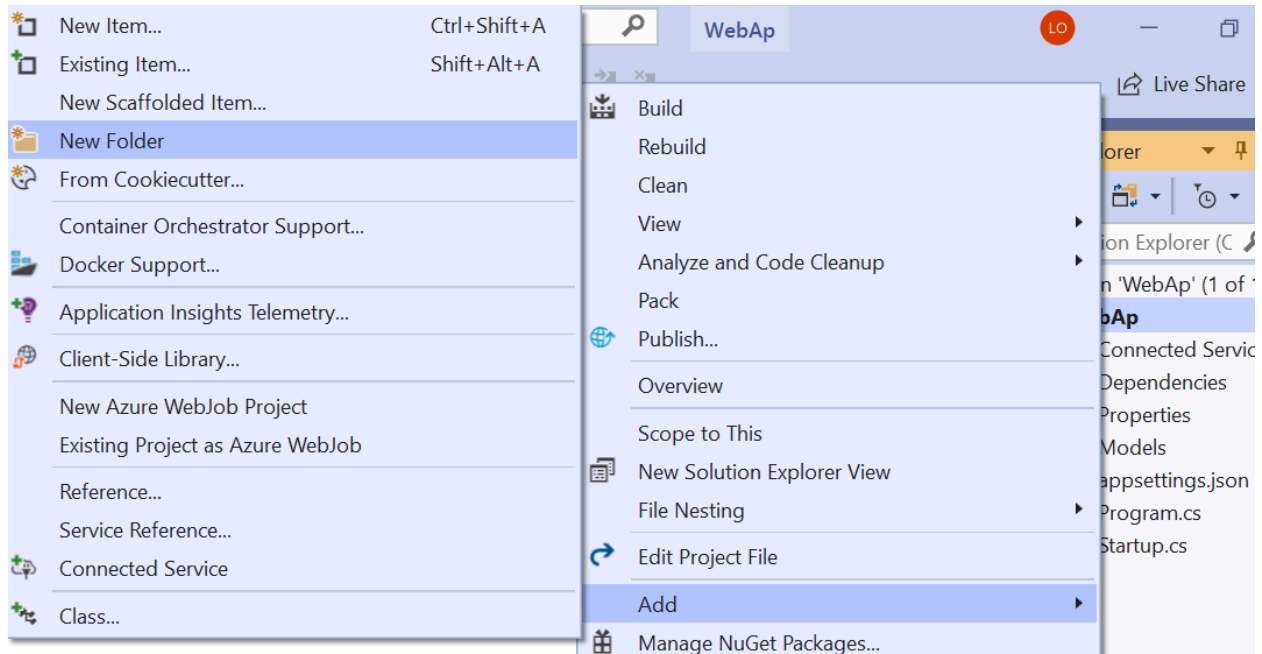
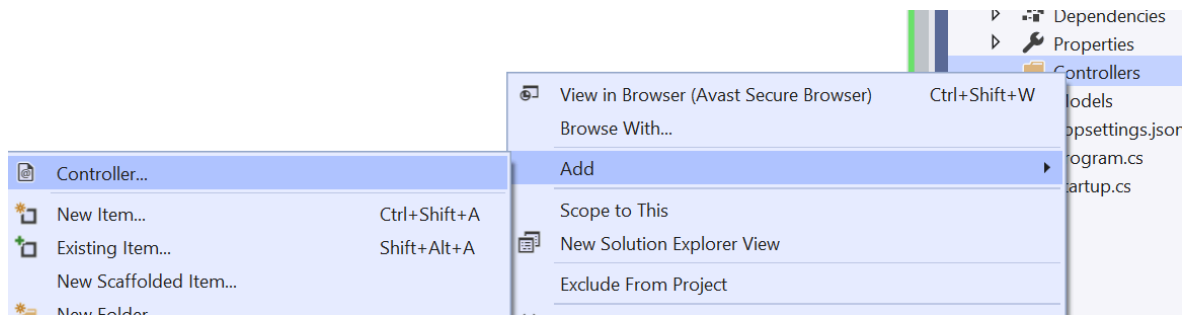


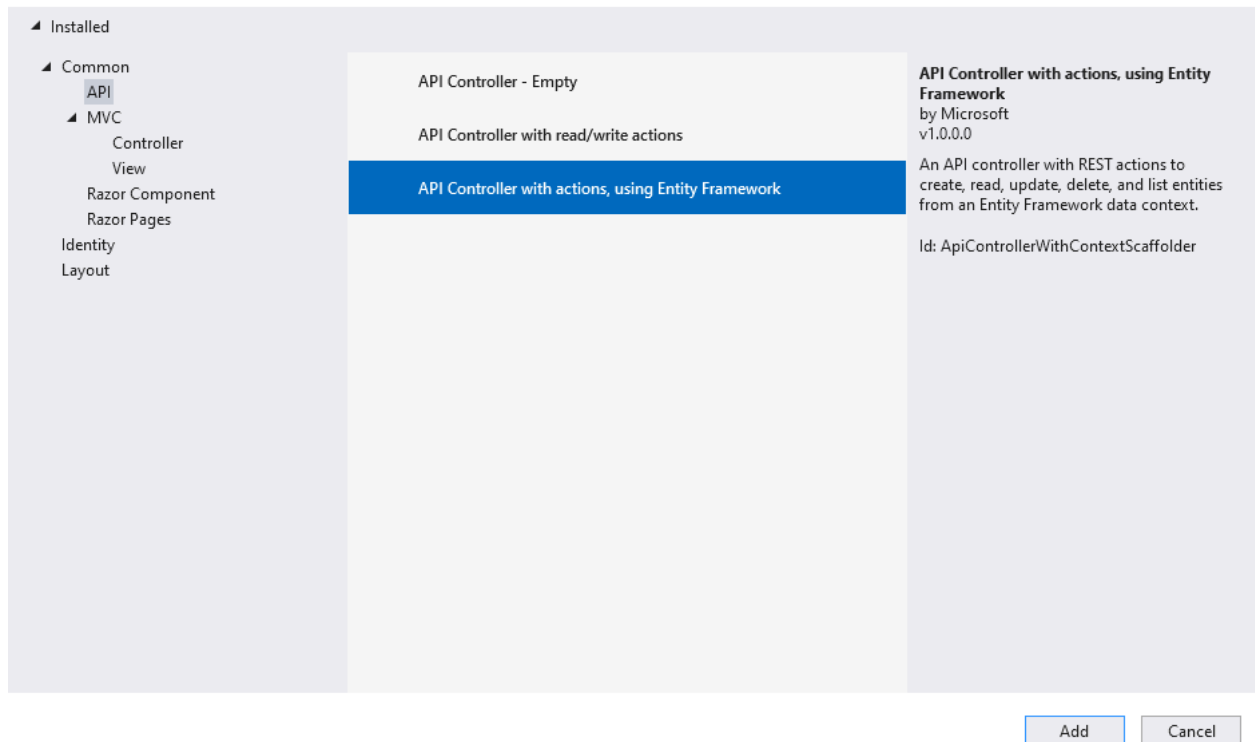
1. Створення контролерів

Далі додамо в проєкт нову папку `Controllers`, а в ній створимо новий арі-контролер для кожного класу моделі. Для цього при додаванні нового елемента в проєкт можна використовувати шаблон `API Controller Class`:





Add New Scaffolded Item

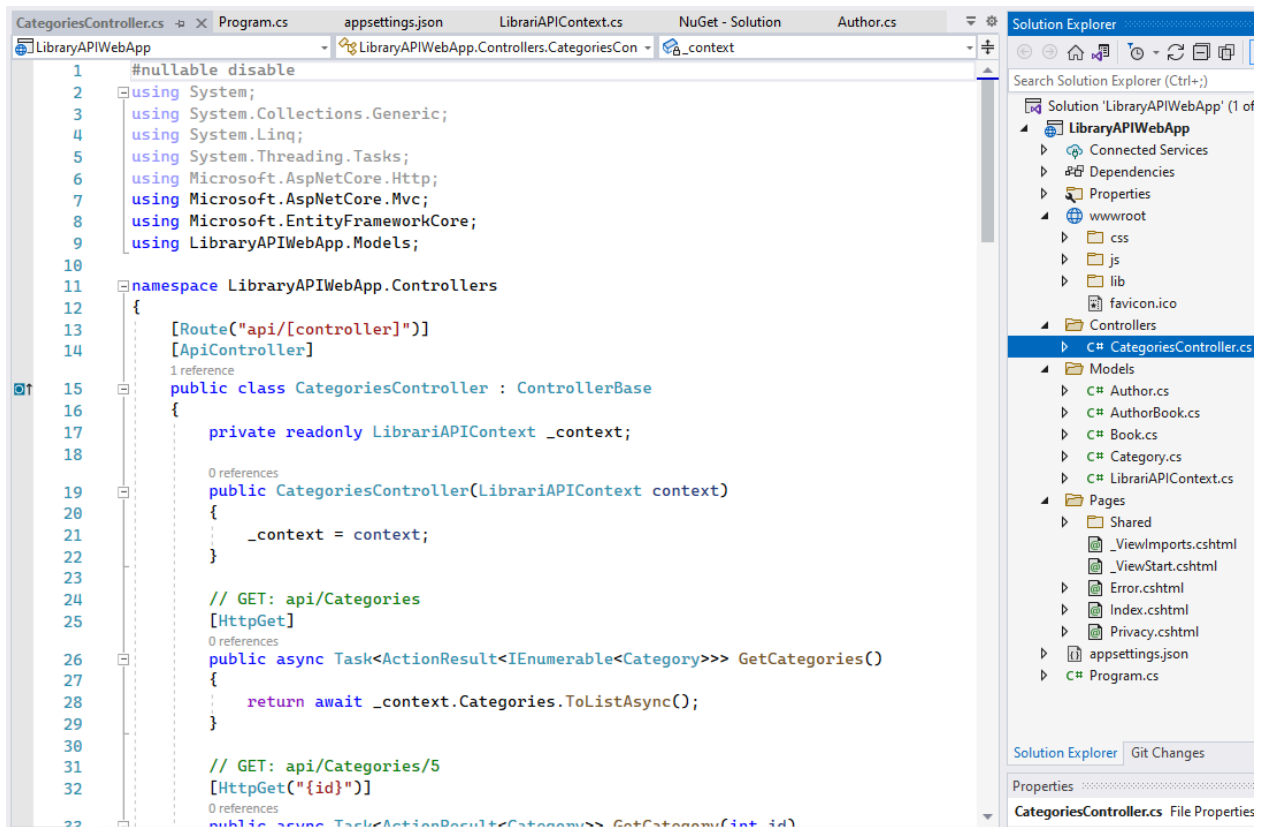


Add API Controller with actions, using Entity Framework

Model class

Data context class

Controller name



До контролера застосовується атрибут `[ApiController]`, який дозволяє використовувати низку додаткових можливостей, зокрема, в плані прив'язки моделі. Також до контролера застосовується атрибут маршрутизації, який вказує, як контролер буде зіставлятися з запитами.

У конструкторі контролера отримуємо контекст даних і використовуємо його для операцій з даними. Також в конструкторі контролера додаємо початкові дані.

Контролер API призначений переважно для обробки запитів протоколу HTTP: Get, Post, Put, Delete, Patch, Head, Options. В даному випадку для кожного типу запитів в контролері визначено свої методи. Так, метод `Get()` обробляє запити типу GET і повертає колекцію об'єктів з БД.

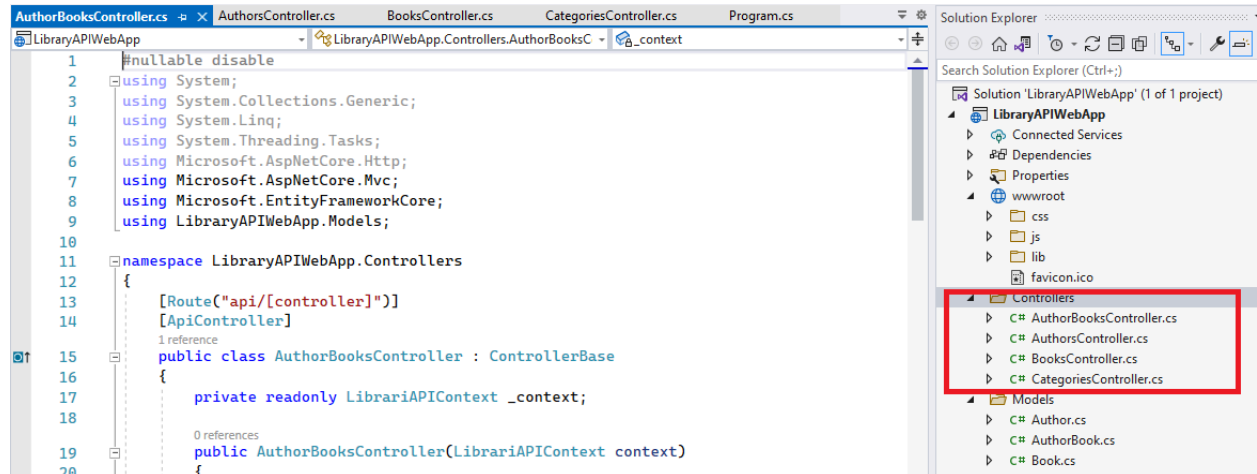
Якщо запит `Get` містить параметр `id` (ідентифікатор об'єкта), то він обробляється іншим методом - `Get(int id)`, який повертає об'єкт по переданому `id`.

Запити типу `Post` обробляються методом `Post`, який отримує з тіла запиту відправлені дані і додає їх в базу даних.

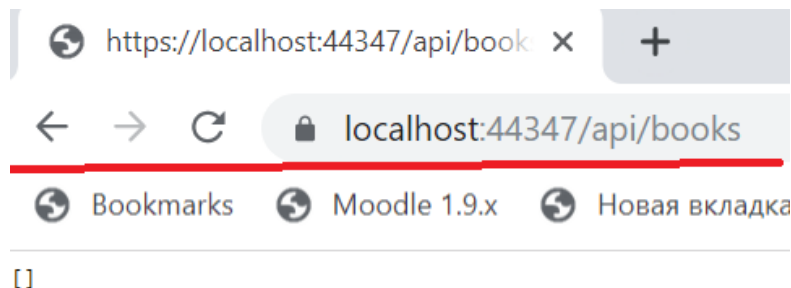
Метод `Put` обробляє запити типу `Put` - отримує дані з запиту і змінює ними об'єкт в базі даних.

І метод Delete (int id) обробляє запити типу Delete, тобто запити на видалення - отримує із запити параметр id і по цим ідентифікатором видаляє об'єкт з БД.

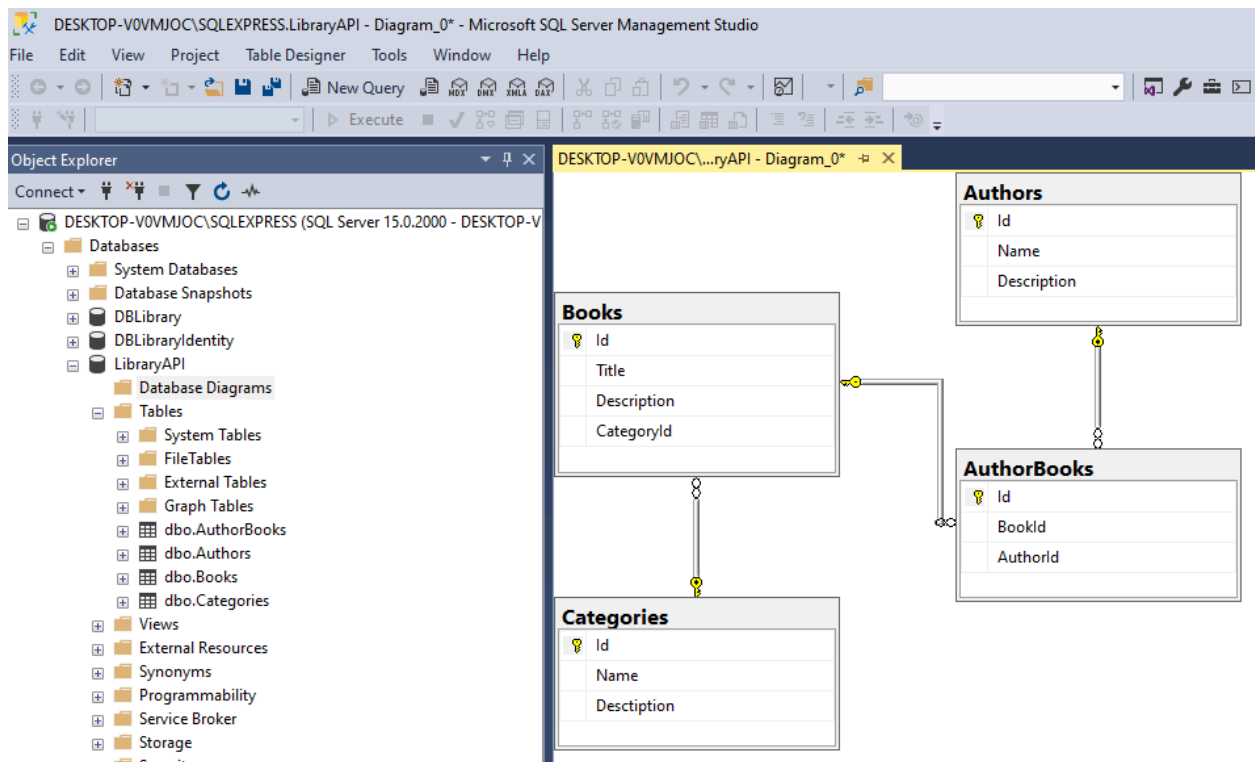
Аналогічно створюємо контролери для усіх класів моделі (за необхідності додаємо методи та/або змінюємо існуючі – це є обов'язковим для здачі етапу на максимальний бал):



Запускаємо:



Ми здійснили звернення до БД, перевіримо її:

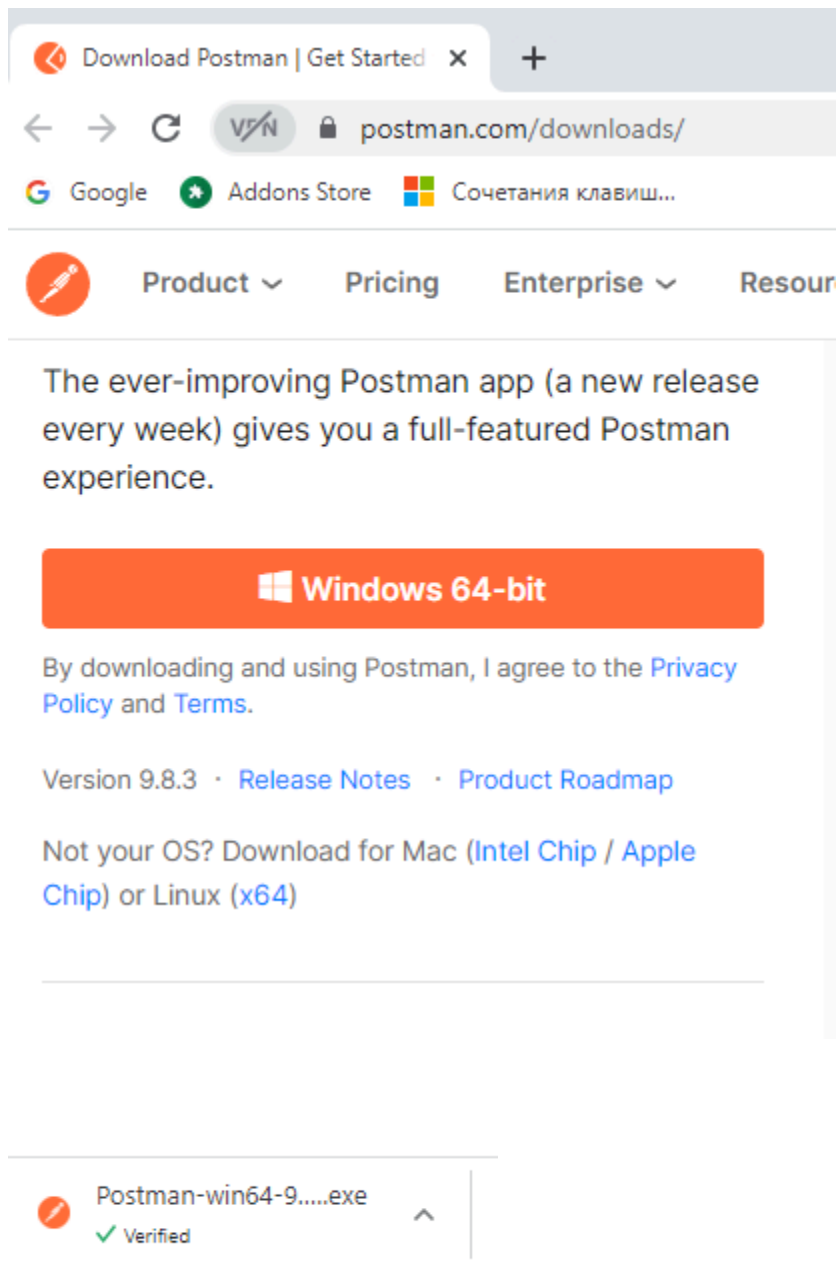


2. Тестування контролерів

Ми створили контролери Web API, і протестували роботу методу GET контролера Books. Однак безпосередньо з рядка браузера крім запитів GET інші типи запитів ми протестувати не можемо. Звичайно, ми можемо створити клієнт у вигляді веб-сторінки, мобільного додатка під якусь платформу або навіть графічного або консольного десктопного додатка, але створення клієнта може зайняти досить багато часу, тоді як нам просто треба протестувати обробку запитів.

Для тестування контролера Web API можна застосовувати спеціальні інструменти, які встановлюються у вигляді окремих додатків, або у вигляді розширень для браузерів, наприклад, Fiddler або Postman.

Встановіть Postman (<https://www.getpostman.com/downloads/>):



Запустіть веб-додаток.

Запустіть Postman (В File-> Settings відключіть параметр **SSL certificate verification** – після завершення роботи увімкніть знову).

Створіть новий запит.

Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

LibraryAPI New Import

Overview GET https://localhost:44361/api/categories

LibraryAPI / https://localhost:44361/api/categories

GET https://localhost:44361/api/categories

Params Auth Headers (6) Body Pre-req. Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (5) Test Results 200 OK 3.28 s 330 B Save Response

Pretty Raw Preview Visualize

```
[{"id":1,"name":"Лорика","description":"111","books":[]}, {"id":2,"name":"Програмування","description":"лодлол","books":[]}]
```

Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

LibraryAPI New Import

Overview GET https://localhost:44361/api/Categories

LibraryAPI / https://localhost:44361/api/Categories

GET https://localhost:44361/api/Categories

Params Auth Headers (8) Body Pre-req. Tests Settings Cookies Beautify

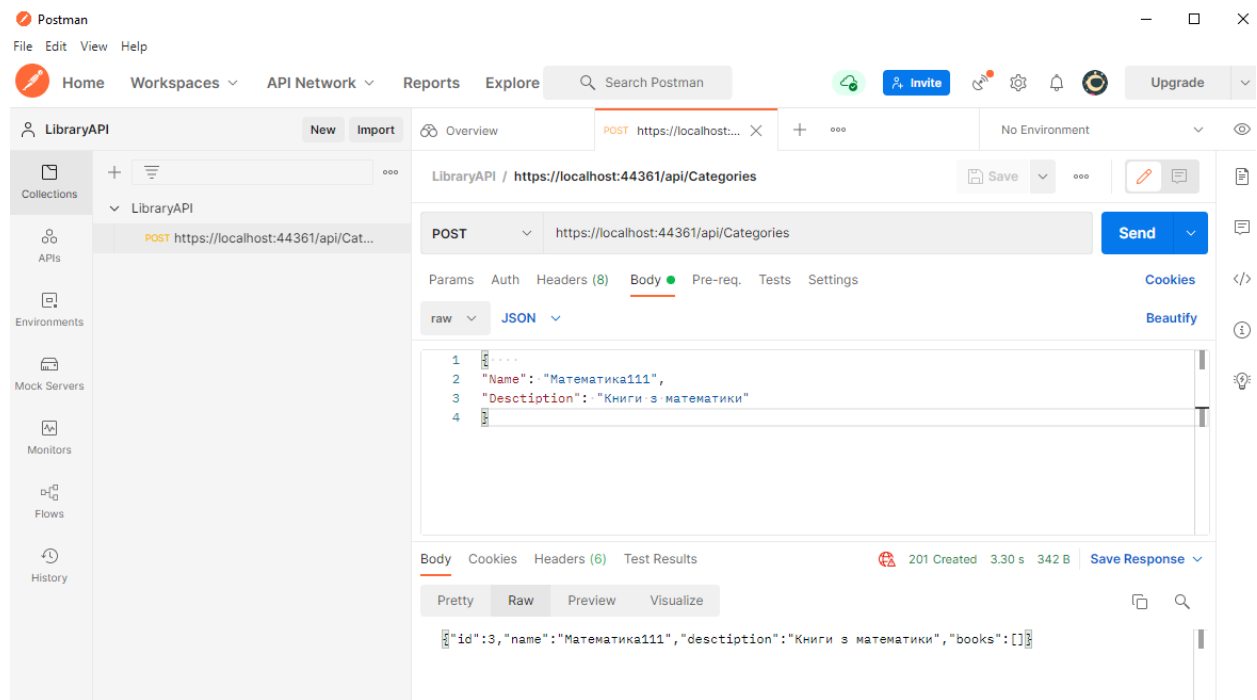
raw JSON

```
1 {
2   "Name": "Математика111",
3   "Description": "Книги з математики"
4 }
```

Body Cookies Headers (5) Test Results 200 OK 2.34 s 434 B Save Response

Pretty Raw Preview Visualize

```
[{"id":1,"name":"Лорика","description":"111","books":[]}, {"id":2,"name":"Програмування","description":"лодлол","books":[]}, {"id":3,"name":"Математика111","description":"Книги з математики","books":[]}]
```



Для здачі цього етапу підготуйте тестові приклади для усіх методів контролерів.