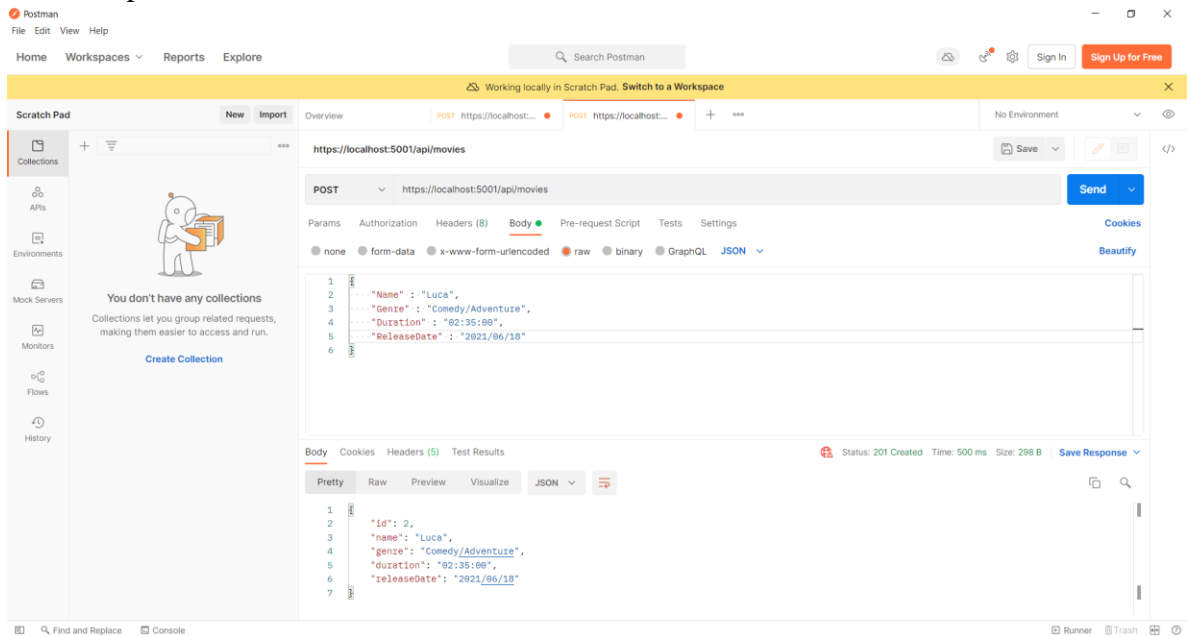


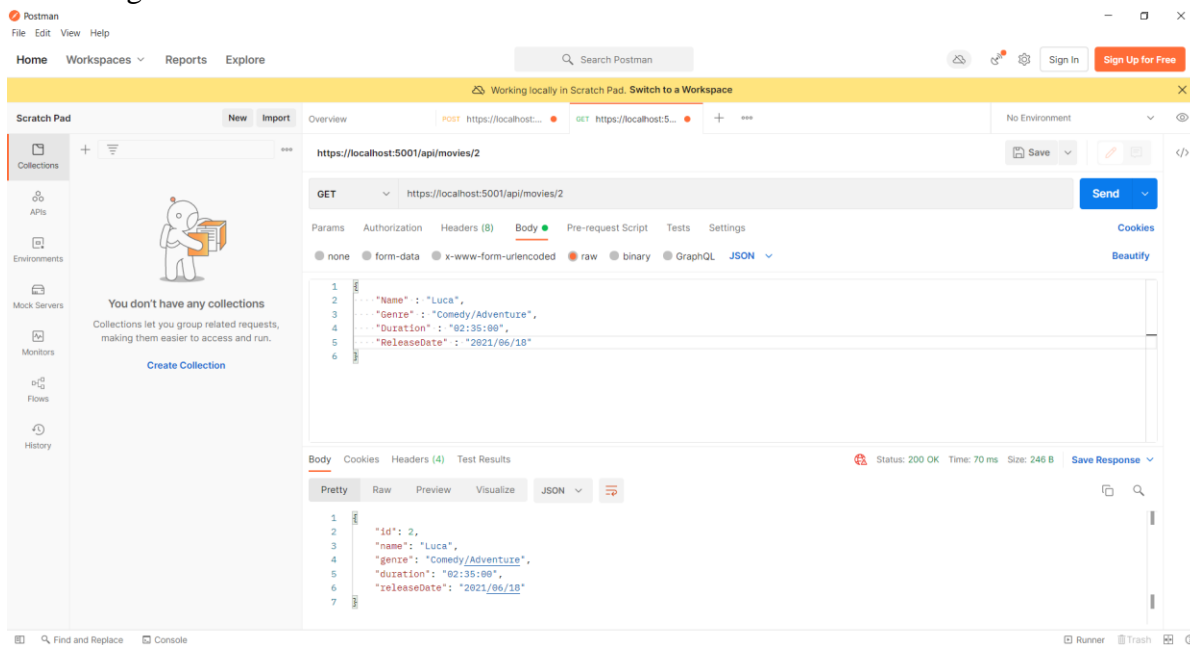
Panduan dan ScreenShot

Panduan:

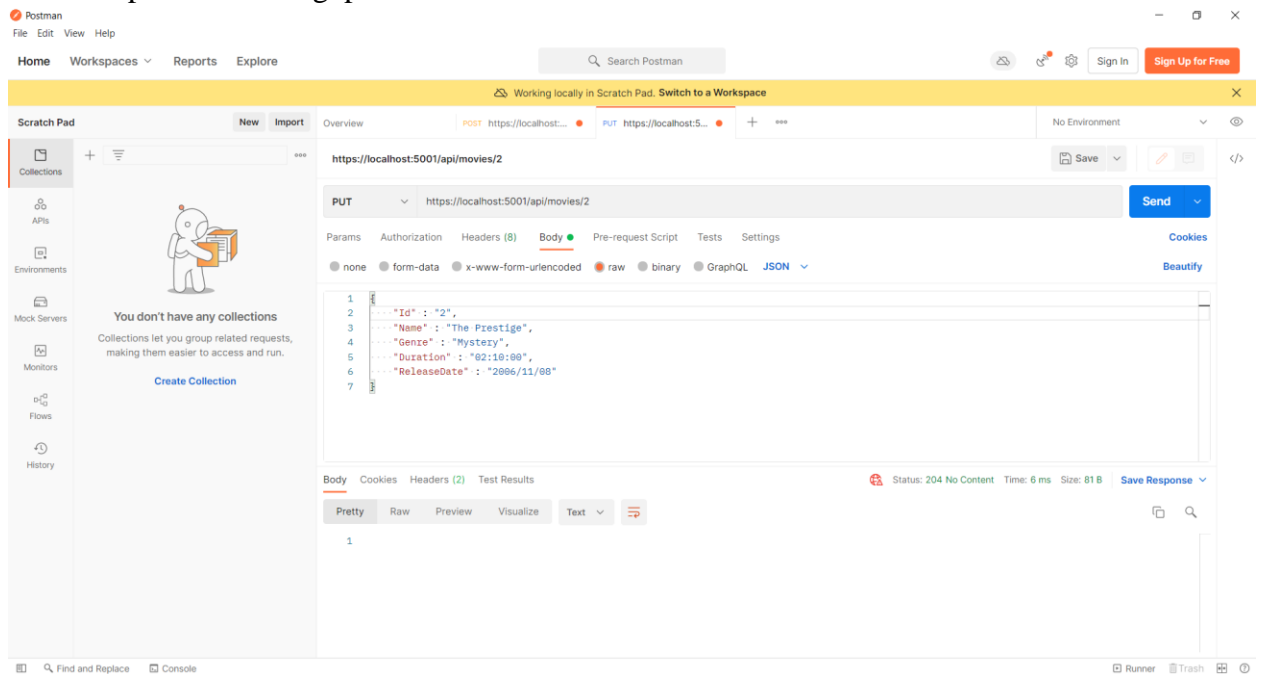
1. Buka projek dan jalankan program dengan dotnet run
2. Buka postman dan masukkan url : <https://localhost:5001/api/movies>
3. Lakukan post untuk insert data ke database



4. Lakukan get untuk melihat isi database



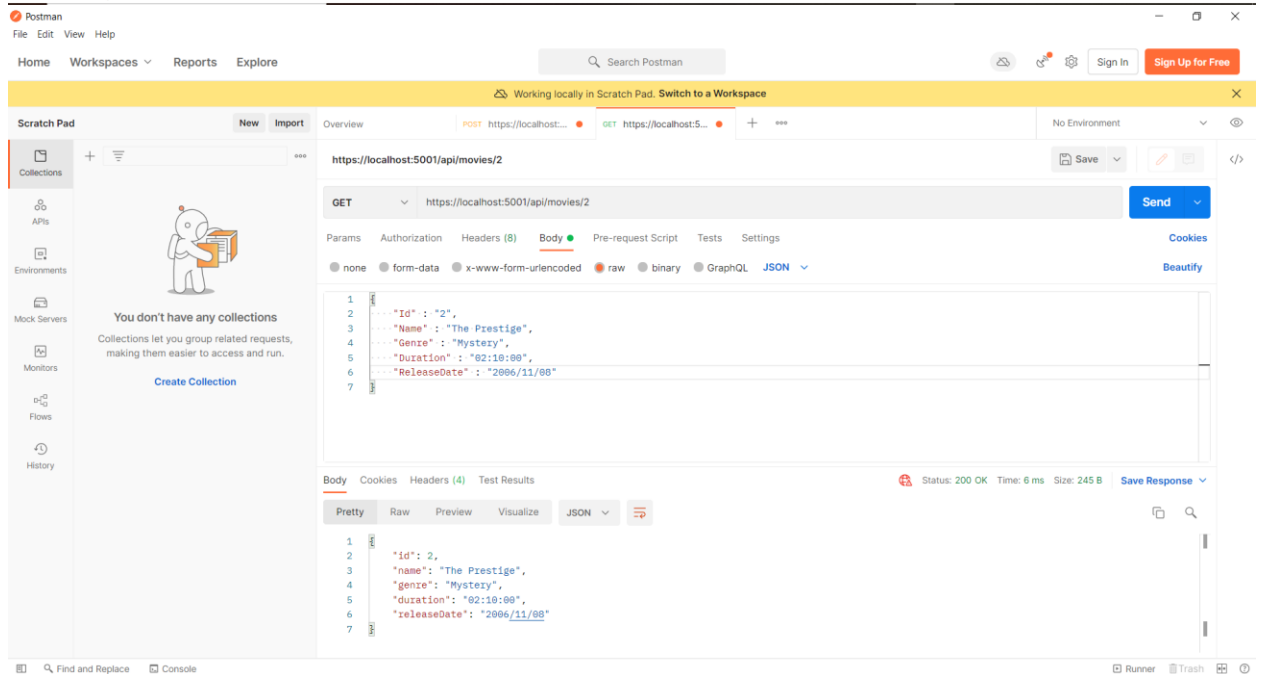
5. Lakukan put untuk mengupdate isi database



Postman interface showing a PUT request to `https://localhost:5001/api/movies/2`. The request body is a JSON object representing movie data:

```
1 {
2   "id": "2",
3   "name": "The Prestige",
4   "genre": "Mystery",
5   "duration": "02:10:00",
6   "releaseDate": "2006/11/08"
7 }
```

The response status is 204 No Content. The interface also shows the 'Body' tab with the request body and the 'Test Results' tab.



Postman interface showing a GET request to `https://localhost:5001/api/movies/2`. The response body is a JSON object representing movie data:

```
1 {
2   "id": 2,
3   "name": "The Prestige",
4   "genre": "Mystery",
5   "duration": "02:10:00",
6   "releaseDate": "2006/11/08"
7 }
```

The response status is 200 OK. The interface also shows the 'Body' tab with the response body and the 'Test Results' tab.

6. Lakukan delete untuk menghapus isi database

The screenshot shows the Postman interface with a DELETE request to `https://localhost:5001/api/movies/2`. The request is configured with the following details:

- Method: DELETE
- URL: `https://localhost:5001/api/movies/2`
- Headers (6): none, form-data, x-www-form-urlencoded, raw, binary, GraphQL
- Body: This request does not have a body

The response is a 200 OK status with a time of 323 ms and a size of 245 B. The response body is displayed in the Pretty view:

```
1 {
2   "id": 2,
3   "name": "The Prestige",
4   "genre": "Mystery",
5   "duration": "02:10:00",
6   "releaseDate": "2006/11/08"
7 }
```

The screenshot shows the Postman interface with a GET request to `https://localhost:5001/api/movies`. The request is configured with the following details:

- Method: GET
- URL: `https://localhost:5001/api/movies`
- Headers (6): none, form-data, x-www-form-urlencoded, raw, binary, GraphQL
- Body: This request does not have a body

The response is a 200 OK status with a time of 7 ms and a size of 255 B. The response body is displayed in the Pretty view:

```
1 {
2   "id": 1,
3   "name": "Tenet",
4   "genre": "Action/Science Fiction",
5   "duration": "02:30:00",
6   "releaseDate": "2021/02/10"
7 }
```