Nama Peserta : Priscilla Gabriela

Kode Peserta : 002 (FSDO001ONL002)

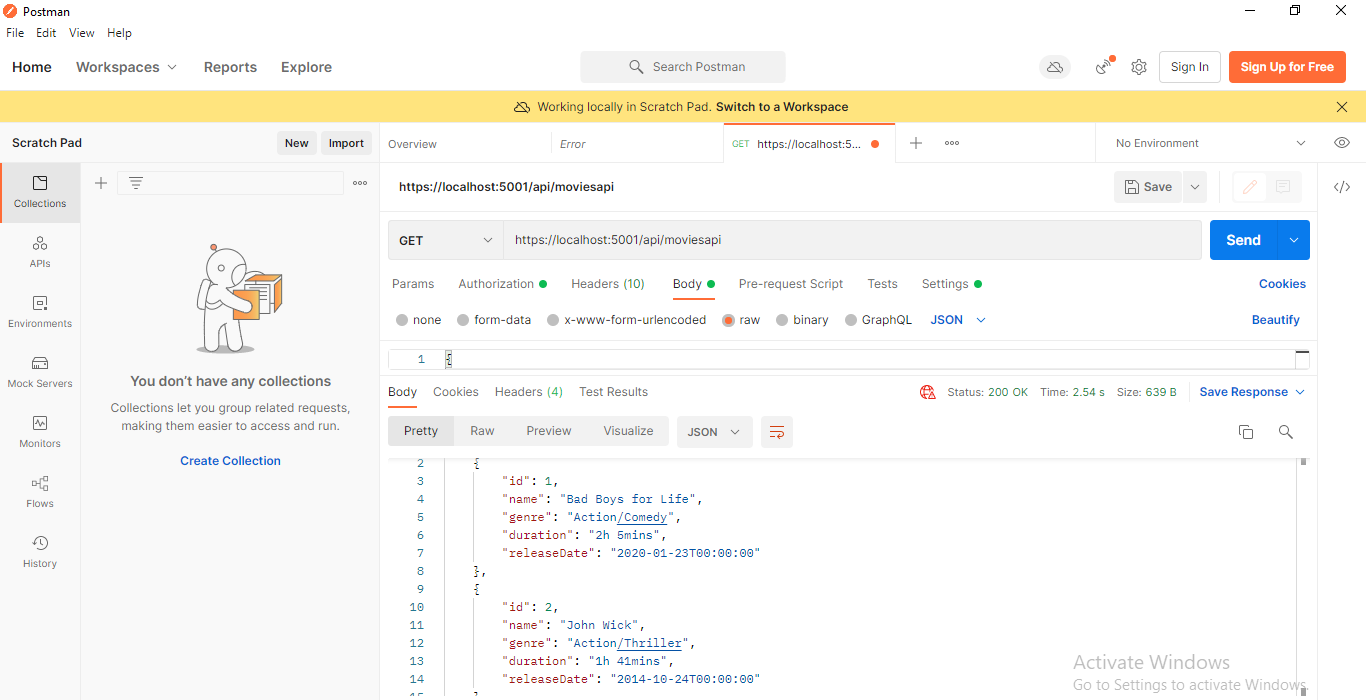
Tugas : Assignment 3

Buat Rest API untuk Database Movie dimana API yang dibuat berfungsi menjadi penghubung ke database dengan method get| post| update| delete records dari dan ke - database MySQL.

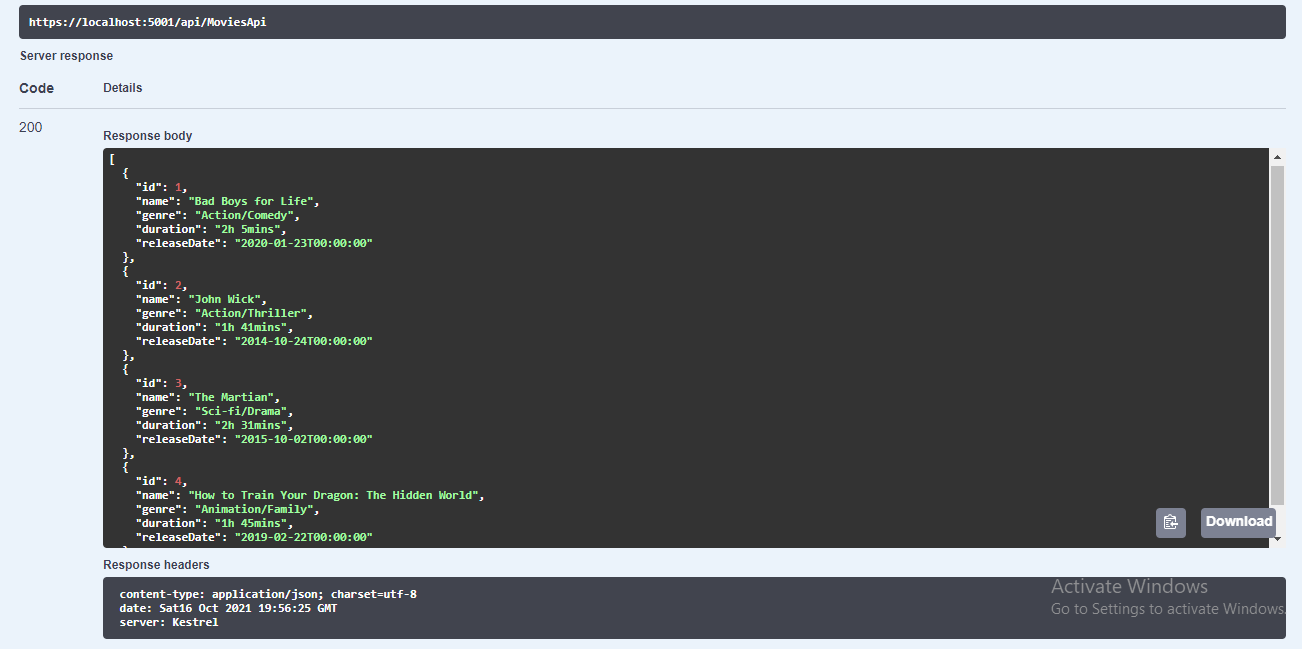
1. API GetMovies [GET]
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method GET
* Menuliskan <https://localhost:5001/api/moviesapi> dan mengklik button send
* Postman akan menampilkan status code dan semua film yang terdapat pada tabel Movie
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan GET /api/MoviesApi, lalu mengklik button execute
* Swagger akan menampilkan status code dan semua film yang terdapat pada tabel Movie

1. Hasil testing dengan Postman



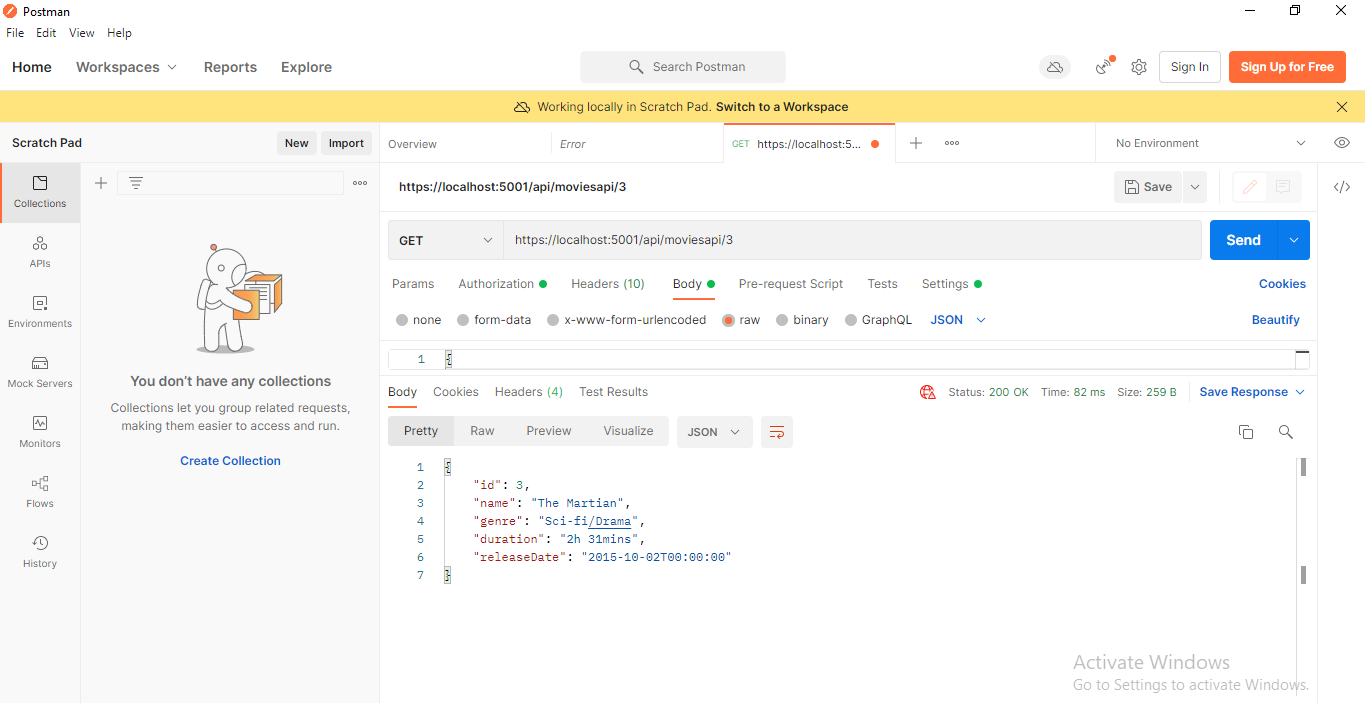
1. Hasil testing dengan Swagger



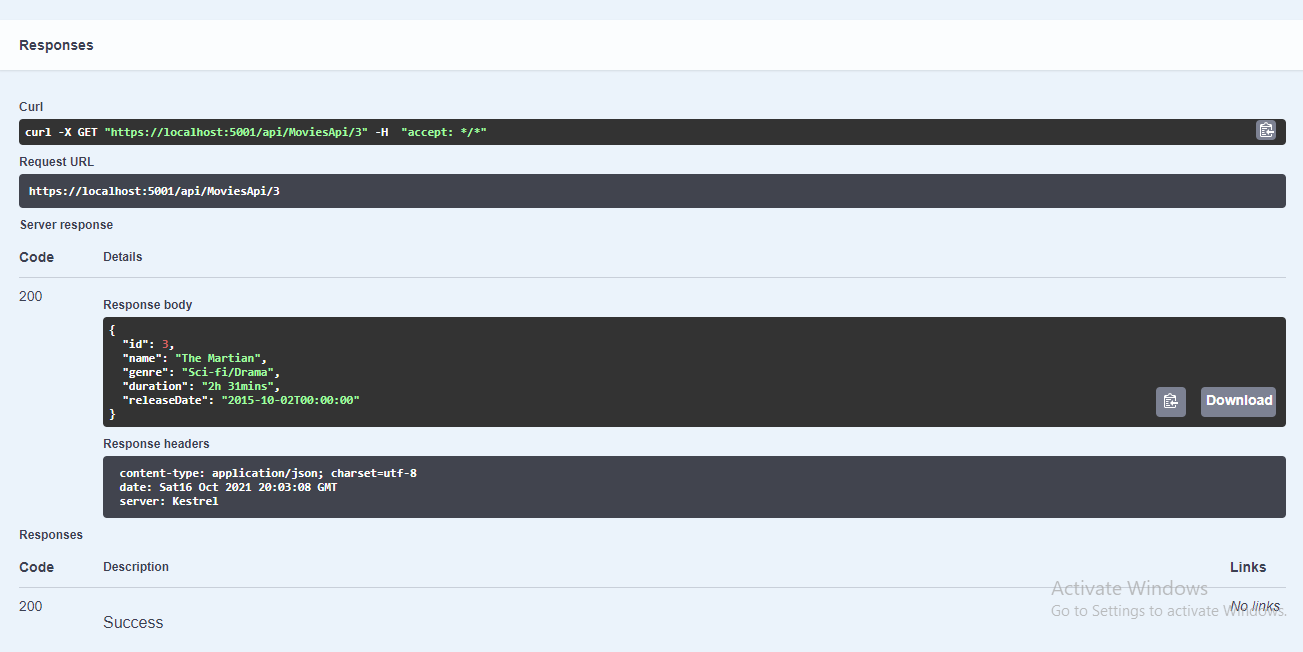
1. API GetMovieById [GET]
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method GET
* Menuliskan <https://localhost:5001/api/moviesapi>/{id} dan mengklik button send
* Postman akan menampilkan status code dan film berdasarkan id yang di-request
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan GET /api/MoviesApi{id}, lalu mengisikan id, dan mengklik button execute
* Swagger akan menampilkan status code dan film berdasarkan id yang di-request

1. Hasil testing dengan Postman



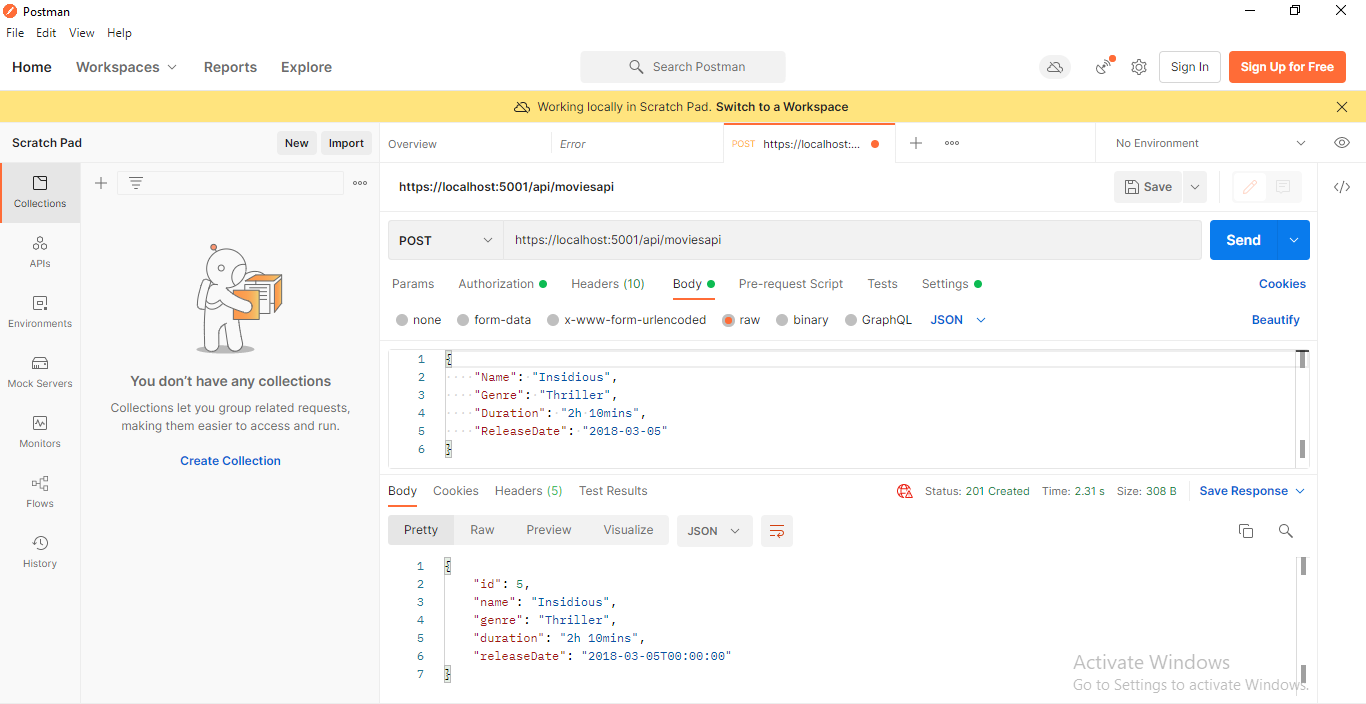
1. Hasil testing dengan Swagger



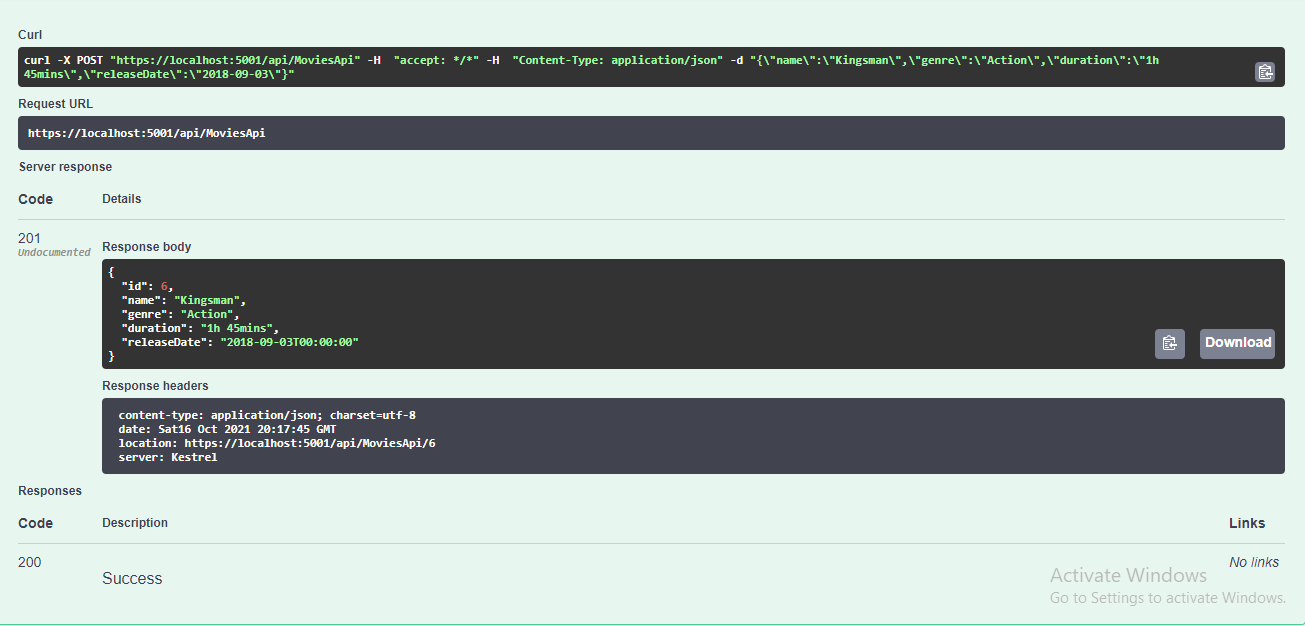
1. API CreateMovie [POST]
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method POST
* Menuliskan [https://localhost:5001/api/moviesapi](https://localhost:5001/api/moviesapi/), mengetik data film yang mau diinput, dan mengklik button send
* Postman akan menampilkan status code dan data film yang baru saja diinput
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan POST /api/MoviesApi, lalu mengisikan data film yang mau diinput, dan mengklik button execute
* Swagger akan menampilkan status code dan data film yang baru saja diinput

1. Hasil testing dengan Postman



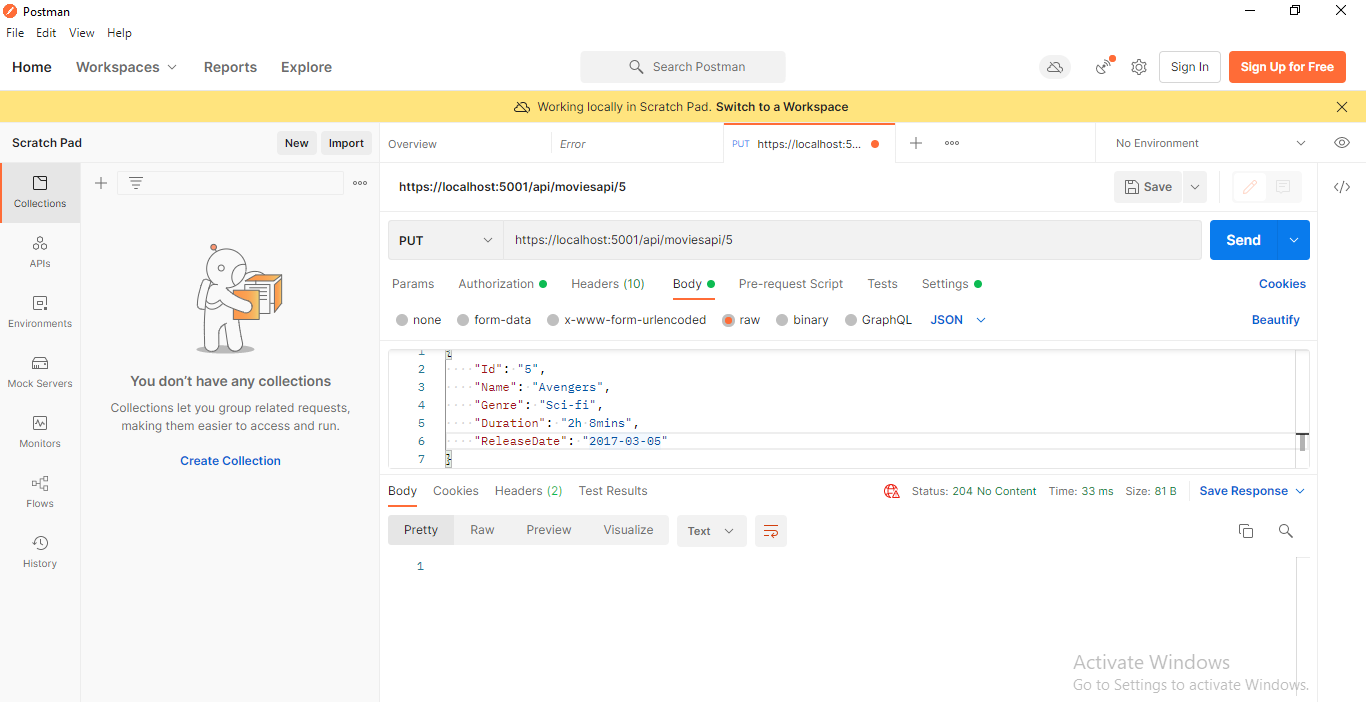
1. Hasil testing dengan Swagger



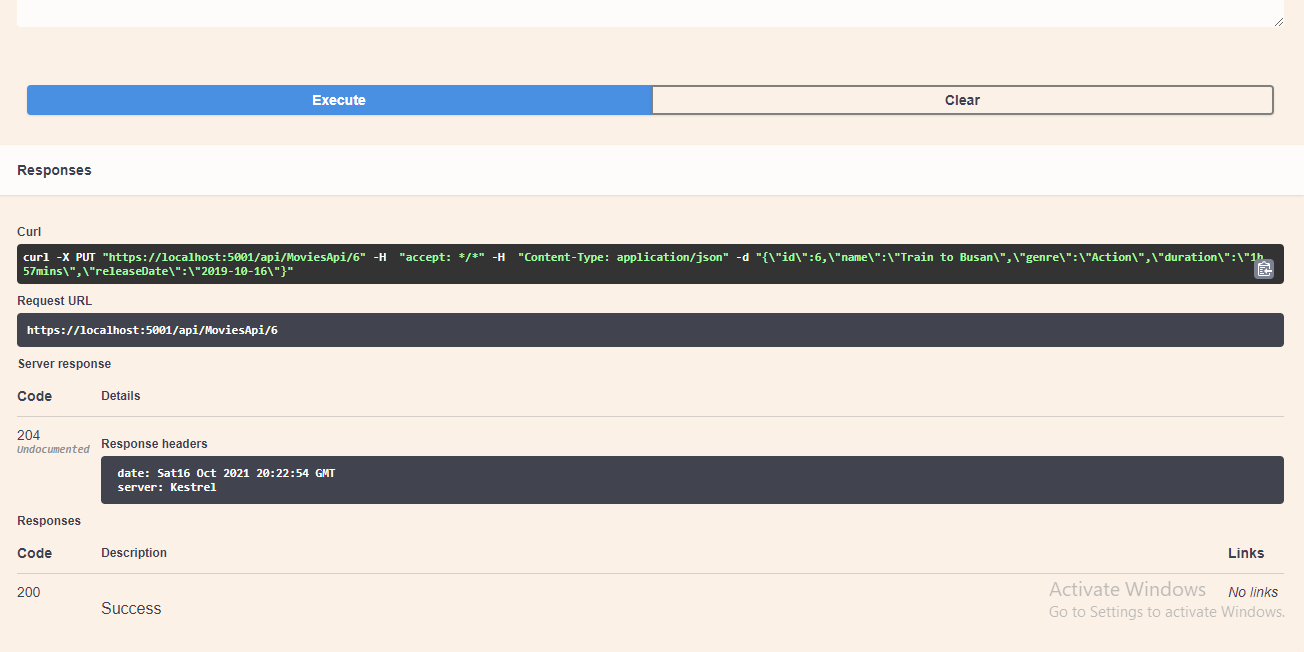
1. API UpdateMovieById [PUT]
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method PUT
* Menuliskan [https://localhost:5001/api/moviesapi](https://localhost:5001/api/moviesapi/)/{id}, mengetik data film yang mau diubah, dan mengklik button send
* Postman akan menampilkan status code dari eksekusi API
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan PUT /api/MoviesApi{id}, lalu mengisikan data film yang mau diubah, dan mengklik button execute
* Swagger akan menampilkan status code dari eksekusi API

1. Hasil testing dengan Postman



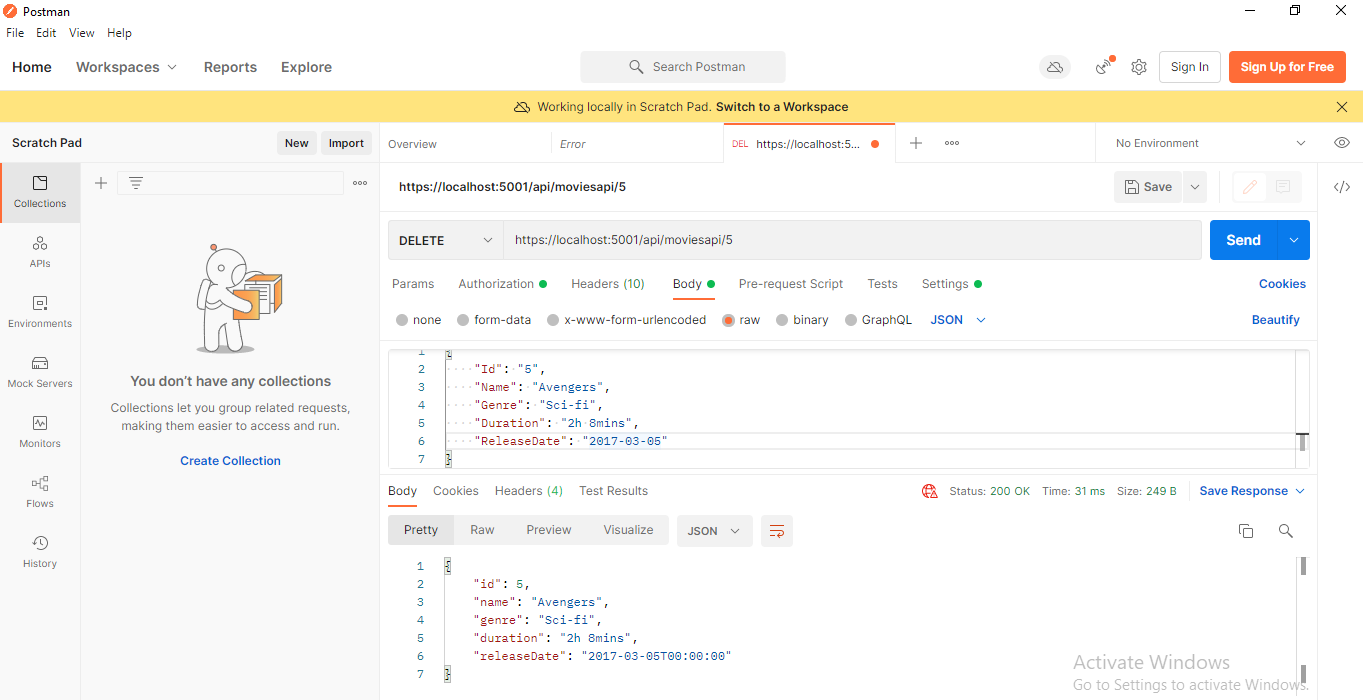
1. Hasil testing dengan Swagger



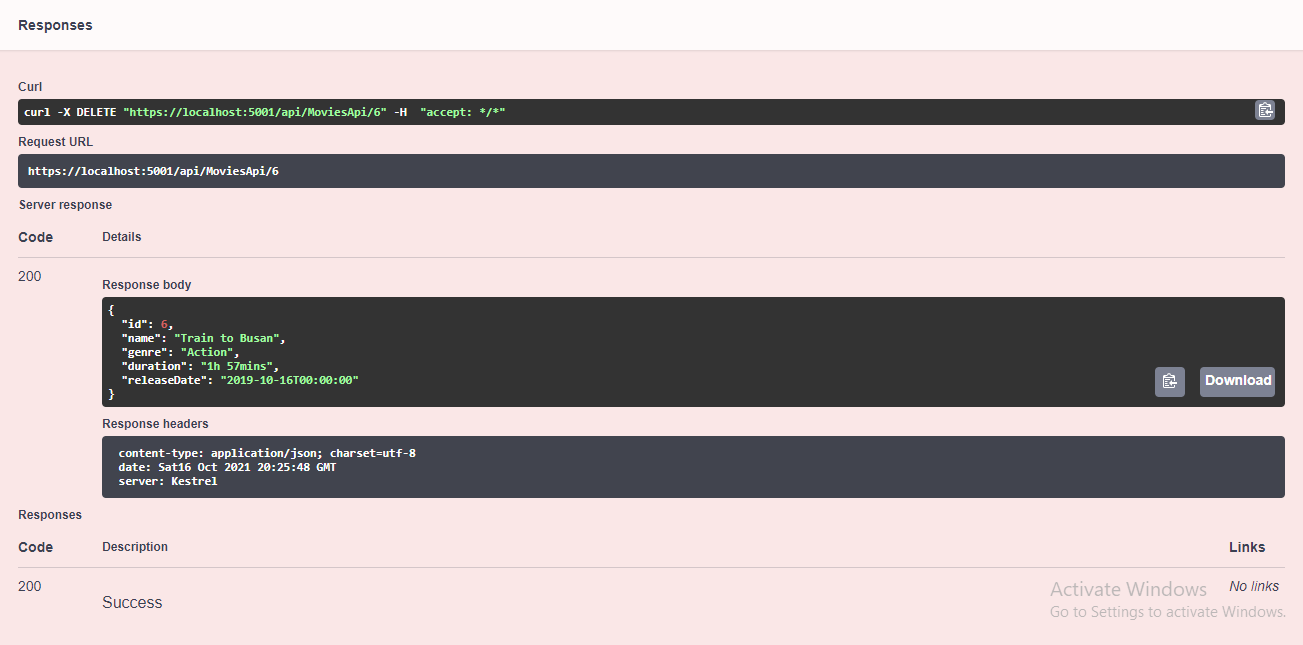
1. API DeleteMovieById [DELETE]
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method DELETE
* Menuliskan [https://localhost:5001/api/moviesapi](https://localhost:5001/api/moviesapi/)/{id} dan mengklik button send
* Postman akan menampilkan status code dan data film yang baru saja dihapus
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan DELETE /api/MoviesApi{id}, lalu mengisikan id film yang mau dihapus, dan mengklik button execute
* Swagger akan menampilkan status code dan data film yang baru saja dihapus

1. Hasil testing dengan Postman



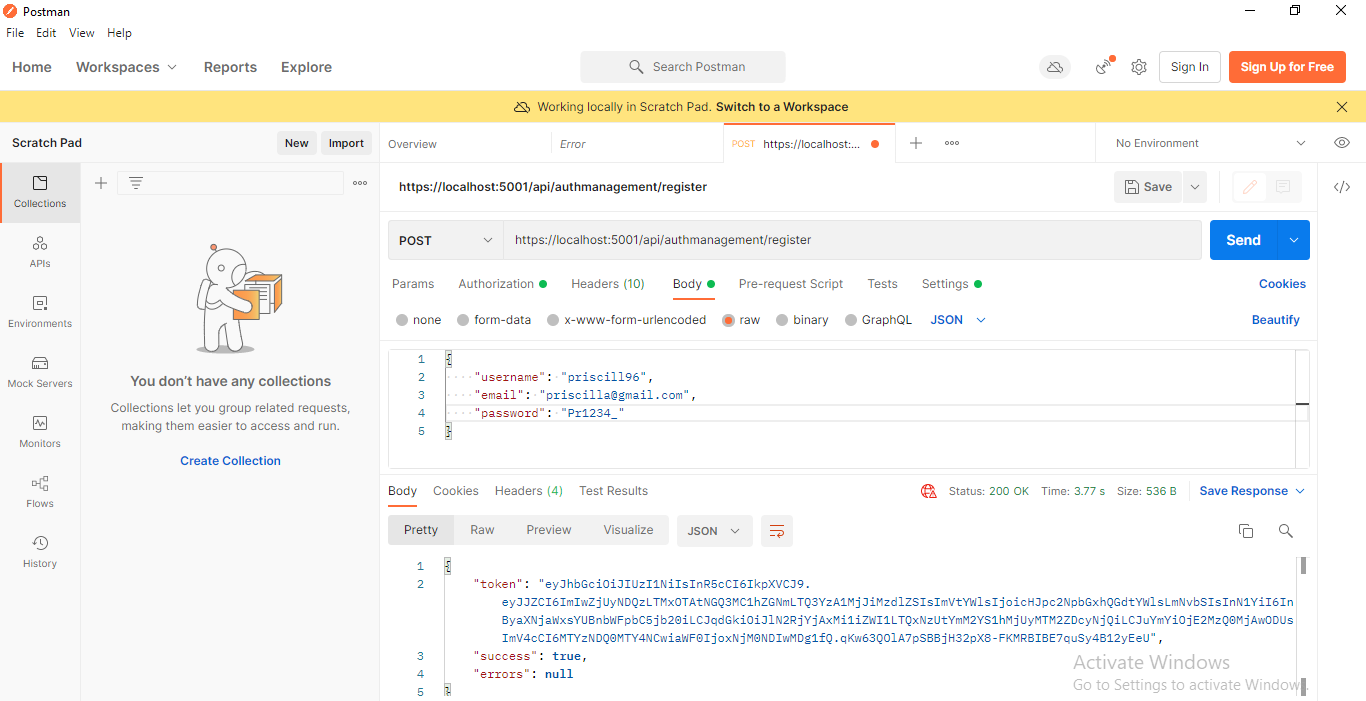
1. Hasil testing dengan Swagger



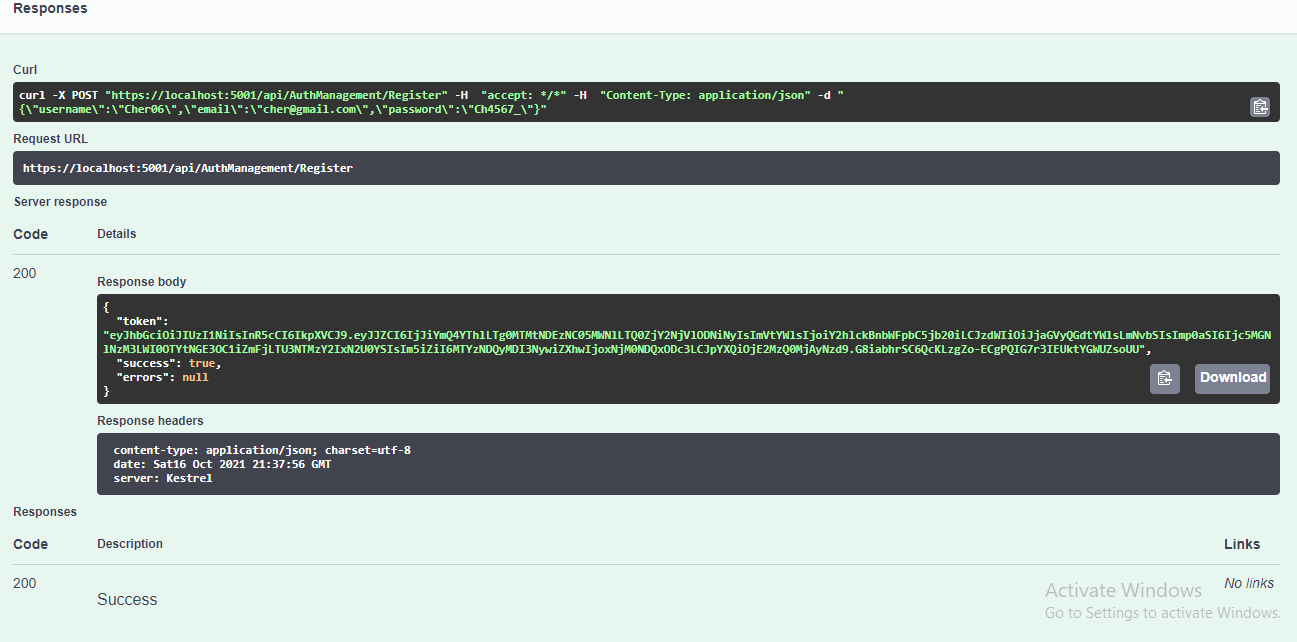
1. JWT Register
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method POST
* Menuliskan <https://localhost:5001/api/authmanagement/register>, mengetik data registrasi, dan mengklik button send
* Postman akan menampilkan status code beserta dengan token
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan POST /api/AuthManagement/Register, lalu mengisikan data registrasi, dan mengklik button execute
* Swagger akan menampilkan status code beserta dengan token

1. Hasil testing dengan Postman



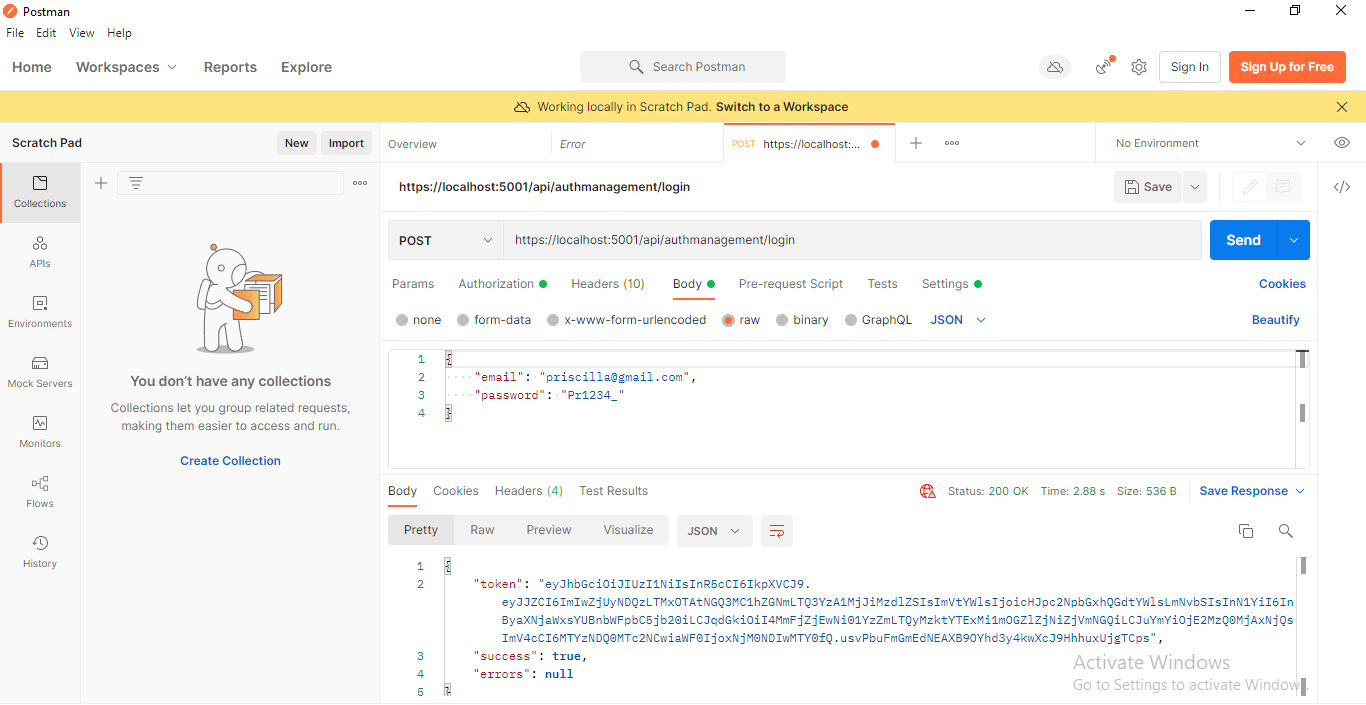
1. Hasil testing dengan Swagger



1. JWT Login
2. Panduan penggunaan

* Membuka project MoviesApi dengan cara klik file🡪open folder, kemudian pilih folder MoviesApi
* Jika ingin melakukan testing API dengan Postman, maka buka cmd sesuai direktori folder dan ketikan dotnet run
* Membuka aplikasi Postman dan memilih method POST
* Menuliskan <https://localhost:5001/api/authmanagement/login>, mengetik data login, dan mengklik button send
* Postman akan menampilkan status code beserta dengan token
* Sementara jika ingin melakukan testing API dengan Swagger, maka buka cmd sesuai direktori folder dan ketikan dotnet watch run
* Halaman browser yang ditujukan untuk testing API dengan Swagger akan ditampilkan
* Mengklik button try it out pada pilihan POST /api/AuthManagement/Login, lalu mengisikan data login, dan mengklik button execute
* Swagger akan menampilkan status code beserta dengan token

1. Hasil testing dengan Postman



1. Hasil testing dengan Swagger

