

This URL documents MovieDB's bearer authentication method:

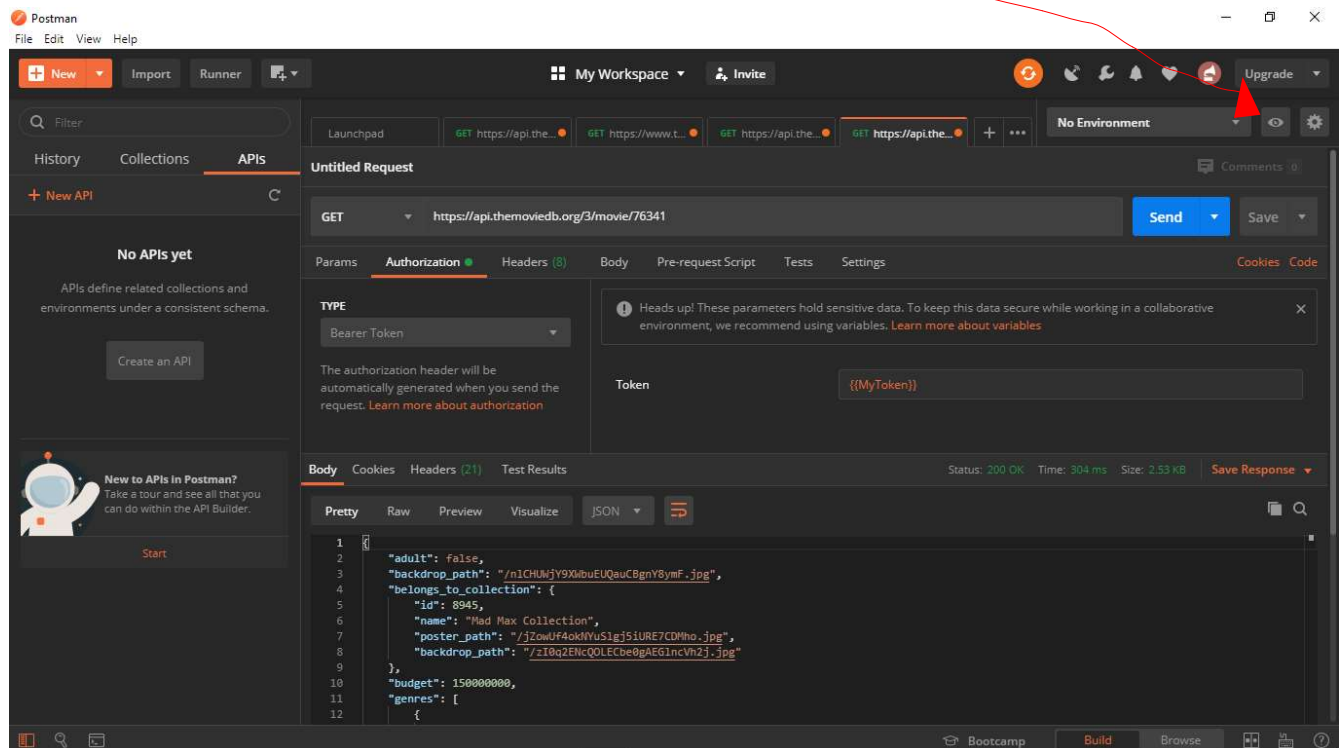
<https://developers.themoviedb.org/3/getting-started/authentication>

The HTTP authorization header for a bearer token must have the token given to you when requested an API key.

Paste your bearer token into your appsettings.json file:

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "ApiKeys": {
    "MovieAPI": "Your API key goes here"
  },
  "ApiBearerTokens": {
    "MovieDbBearerToken": "Your Bearer Token goes here"
  },
  "ConnectionStrings": {
    "DefaultConnection": "Server=.;Database=MovieDb;Trusted_Connection=True;"
  }
}
```

In postman, verify the authorization header with a bearer token as sent to the API. The Token used in the image is a variable I called MyToken setup in the “eye” button:



In the DAL class in the app, I changed the constructor and two methods:

```
using Microsoft.Extensions.Configuration;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net.Http;
using System.Net.Http.Headers;
using System.Text.Json;
using System.Threading.Tasks;

namespace GC_Capstone5.Models
{
    public class MovieDAL
    {
        private readonly string _apiKey;
        private readonly string _apiBearerToken;

        public MovieDAL(IConfiguration configuration)
        {
            _apiKey = configuration.GetSection("ApiKeys")["MovieAPI"];
            _apiBearerToken = configuration.GetSection("ApiBearerTokens")["MovieDbBearerToken"];
        }

        public HttpClient GetClient()
        {
            var client = new HttpClient();
            client.BaseAddress = new Uri("https://api.themoviedb.org");
            client.DefaultRequestHeaders.Authorization =
                new System.Net.Http.Headers.AuthenticationHeaderValue("Bearer", _apiBearerToken);
            return client;
        }

        public async Task<string> GetRawJSON(int movieId)
        {
            var client = GetClient();
            var response = await client.GetAsync($" /3/movie/{movieId}?api_key={_apiKey}");
            var movieJSON = await response.Content.ReadAsStringAsync();
            return movieJSON;
        }

        public async Task<Movie> GetMovie(int movieId)
        {
            var client = GetClient();
            var response = await client.GetAsync($" /3/movie/{movieId}");
            // var response = await client.GetAsync($" /3/movie/{movieId}?api_key={_apiKey}");
            var movieJSON = await response.Content.ReadAsStringAsync();
            Movie movie = JsonSerializer.Deserialize<Movie>(movieJSON);
            return movie;
        }
    }
}
```

And finally, the Controller was changed

- 1) to move the API key to the DAL (The key and bearer token are details only the DAL should privately know about.)
- 2) in the Controller constructor, pass the Configuration object to the DAL constructor