

[ASP.NET Core 1.1](#) ▾

Version

2.2 Preview 2

2.1

2.0

1.1

1.0

Work with SQLite in an ASP.NET Core MVC app

📅 04/07/2017 ⌚ 2 minutes to read Contributors    

In this article

[SQLite](#)

[Seed the database](#)

By [Rick Anderson](#)

The `MvcMovieContext` object handles the task of connecting to the database and mapping `Movie` objects to database records. The database context is registered with the [Dependency Injection](#) container in the `ConfigureServices` method in the *Startup.cs* file:

C#

 Copy

```
public void ConfigureServices(IServiceCollection services)
{
    // Add framework services.
    services.AddMvc();

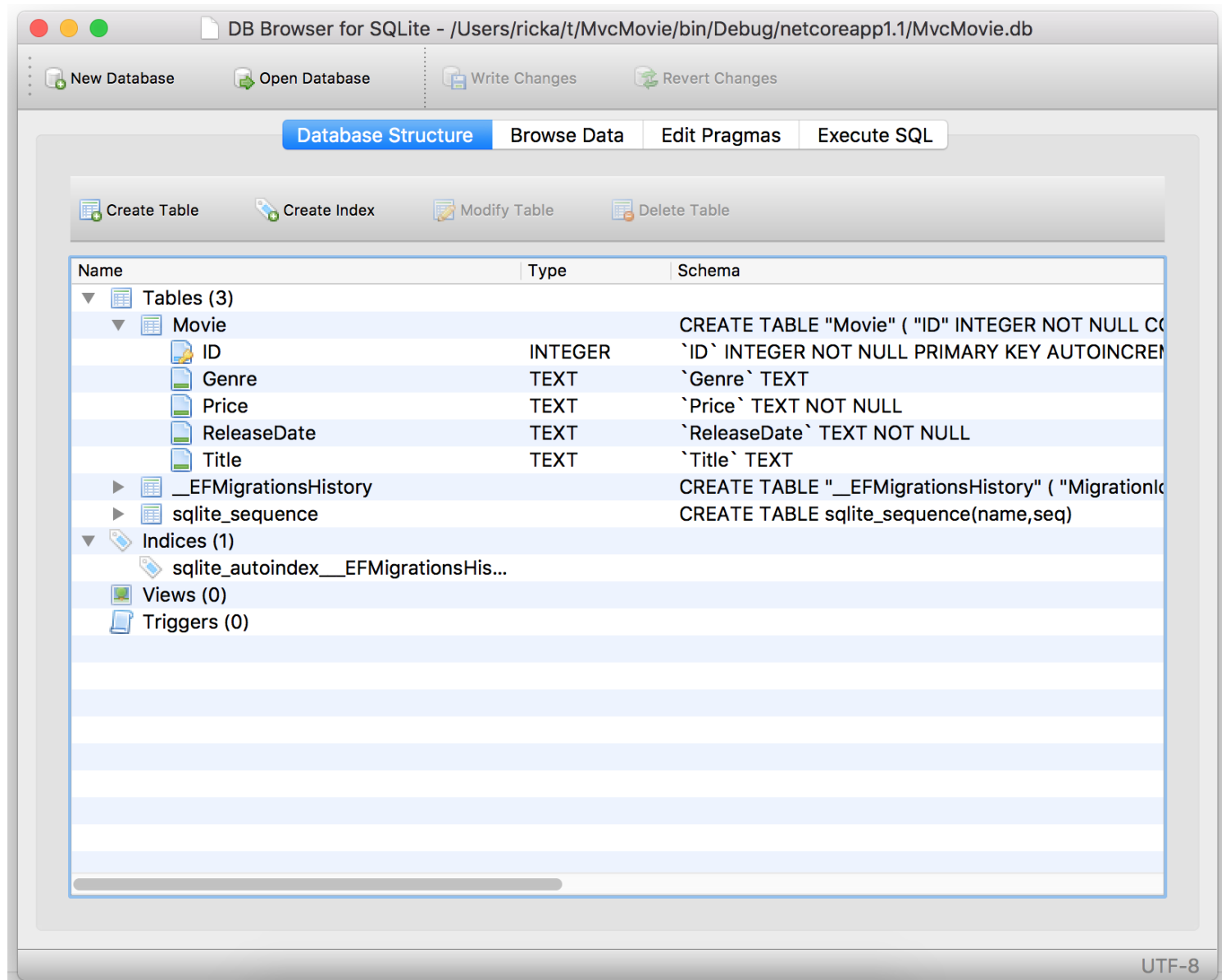
    services.AddDbContext<MvcMovieContext>(options =>
        options.UseSqlite("Data Source=MvcMovie.db"));
}
```

SQLite

The [SQLite](#) website states:

SQLite is a self-contained, high-reliability, embedded, full-featured, public-domain, SQL database engine. SQLite is the most used database engine in the world.

There are many third party tools you can download to manage and view a SQLite database. The image below is from [DB Browser for SQLite](#). If you have a favorite SQLite tool, leave a comment on what you like about it.



Seed the database

Create a new class named `SeedData` in the *Models* folder. Replace the generated code with the following:

C# Copy

```
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.DependencyInjection;
using System;
using System.Linq;

namespace MvcMovie.Models
```

```

{
    public static class SeedData
    {
        public static void Initialize(IServiceProvider serviceProvider)
        {
            using (var context = new MvcMovieContext(
                serviceProvider.GetRequiredService<DbContextOptions<MvcMovieContext>>()))
            {
                // Look for any movies.
                if (context.Movie.Any())
                {
                    return; // DB has been seeded
                }

                context.Movie.AddRange(
                    new Movie
                    {
                        Title = "When Harry Met Sally",
                        ReleaseDate = DateTime.Parse("1989-1-11"),
                        Genre = "Romantic Comedy",
                        Price = 7.99M
                    },

                    new Movie
                    {
                        Title = "Ghostbusters ",
                        ReleaseDate = DateTime.Parse("1984-3-13"),
                        Genre = "Comedy",
                        Price = 8.99M
                    },

                    new Movie
                    {
                        Title = "Ghostbusters 2",
                        ReleaseDate = DateTime.Parse("1986-2-23"),
                        Genre = "Comedy",
                        Price = 9.99M
                    },

                    new Movie
                    {
                        Title = "Rio Bravo",
                        ReleaseDate = DateTime.Parse("1959-4-15"),
                        Genre = "Western",
                        Price = 3.99M
                    }
                );
                context.SaveChanges();
            }
        }
    }
}

```

If there are any movies in the DB, the seed initializer returns.

C#

 Copy

```
if (context.Movie.Any())
{
    return;    // DB has been seeded.
}
```

Add the seed initializer

Add the seed initializer to the `Main` method in the *Program.cs* file:

C#

 Copy

```
using Microsoft.AspNetCore;
using Microsoft.AspNetCore.Hosting;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Logging;
using MvcMovie.Models;
using System;

namespace MvcMovie
{
    public class Program
    {
        public static void Main(string[] args)
        {
            var host = BuildWebHost(args);

            using (var scope = host.Services.CreateScope())
            {
                var services = scope.ServiceProvider;

                try
                {
                    // Requires using MvcMovie.Models;
                    SeedData.Initialize(services);
                }
                catch (Exception ex)
                {
                    var logger = services.GetRequiredService<ILogger<Program>>();
                    logger.LogError(ex, "An error occurred seeding the DB.");
                }
            }

            host.Run();
        }

        public static IWebHost BuildWebHost(string[] args) =>
```

```

        WebHost.CreateDefaultBuilder(args)
            .UseStartup<Startup>()
            .Build();
    }
}

```

Test the app

Delete all the records in the DB (So the seed method will run). Stop and start the app to seed the database.

The app shows the seeded data.

MvcMovie

[Create New](#)

Title	ReleaseDate	Genre	Price	
When Harry Met Sally	1/11/1989 12:00:00 AM	Romantic Comedy	7.99	Edit Details Delete
Ghostbusters	3/13/1984 12:00:00 AM	Comedy	8.99	Edit Details Delete
Ghostbusters 2	2/23/1986 12:00:00 AM	Comedy	9.99	Edit Details Delete
Rio Bravo	4/15/1959 12:00:00 AM	Western	3.99	Edit Details Delete

© 2017 - MvcMovie