

# How to Create a Database Application (Create,Read,Update,Delete) CRUD Application using Scaffolding Framework in Visual Studio Community Edition 2022

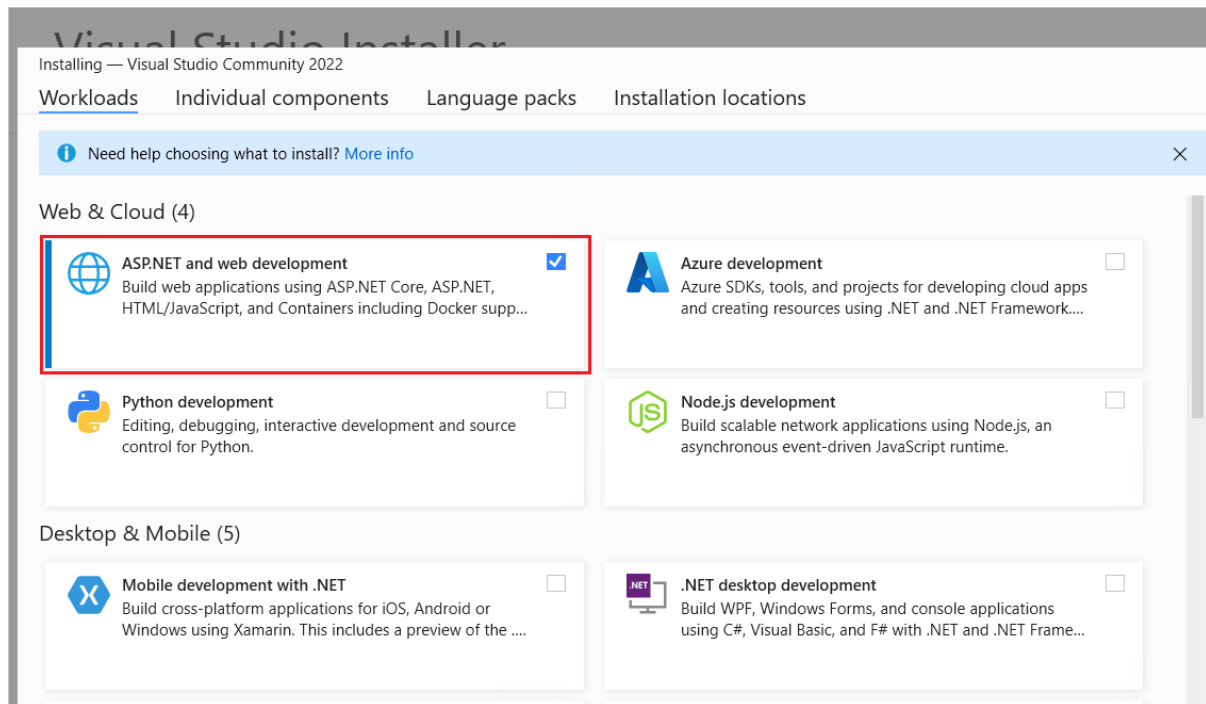
Raman Deep Singh

Scaffolding is a framework or you can say a technique through which you can database applications in Asp.net core with writing minimal or very less code.

It just requires a few clicks and writing very less code and you can create CRUD Database Applications (Create Read Update Delete) Applications in Asp.net Core

Pre Requirements or Prerequisites are

You are having asp.net and web development workload in visual studio 2022 community edition



Next Step is create a Razor Pages Web App

- Start Visual Studio and select **Create a new project**.
- In the **Create a new project** dialog, select **ASP.NET Core Web App > Next**.
- In the **Configure your new project** dialog, enter RazorPagesMovie for **Project name**. It's important to name the project **RazorPagesMovie**, including matching the capitalization, so the namespaces will match when you copy and paste example code.
- Select **Next**.
- In the **Additional information** dialog:
  - Select **.NET 7.0**.
  - Verify: **Do not use top-level statements** is unchecked.
- Select **Create**.

Additional information

ASP.NET Core Web AppC#LinuxmacOSWindowsCloudServiceWeb

Framework ⓘ

.NET 7.0 (Standard Term Support)

Authentication type ⓘ

None

☒ Configure for HTTPS ⓘ

☐ Enable Docker ⓘ

Docker QS ⓘ

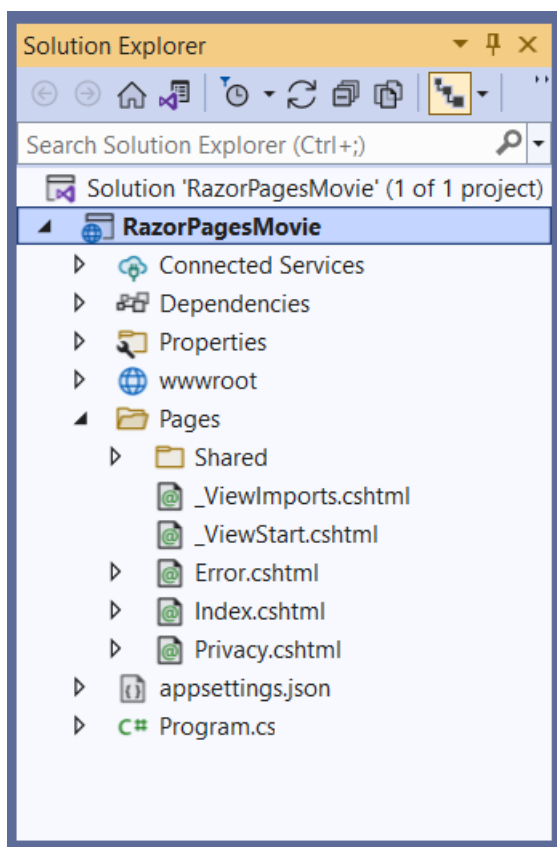
Linux

☐ Do not use top-level statements ⓘ

Back

Create

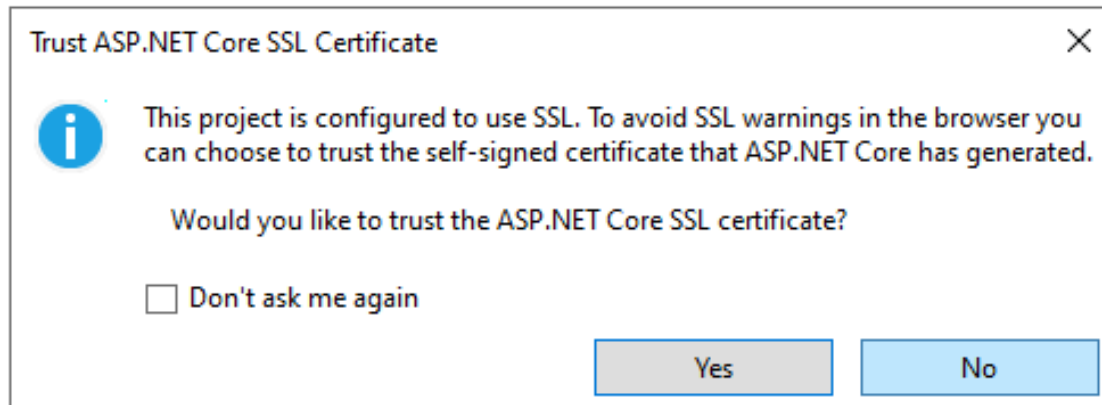
The following starter project is created:



Run The App

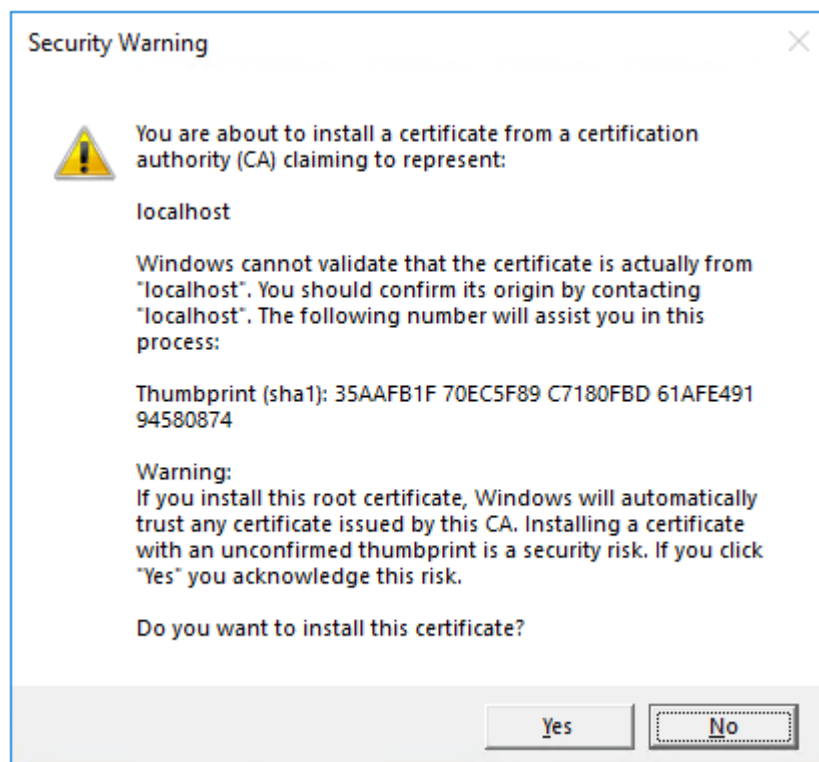
Select **RazorPagesMovie** in **Solution Explorer**, and then press Ctrl+F5 to run without the debugger.

Visual Studio displays the following dialog when a project is not yet configured to use SSL:



Select **Yes** if you trust the IIS Express SSL certificate.

The following dialog is displayed:



Select **Yes** if you agree to trust the development certificate.

# add a model to a Razor Pages app in ASP.NET Core

1. In **Solution Explorer**, right-click the *RazorPagesMovie* project > **Add** > **New Folder**. Name the folder *Models*.
2. Right-click the *Models* folder. Select **Add** > **Class**. Name the class *Movie*.
3. Add the below written code to *Movie.cs* file

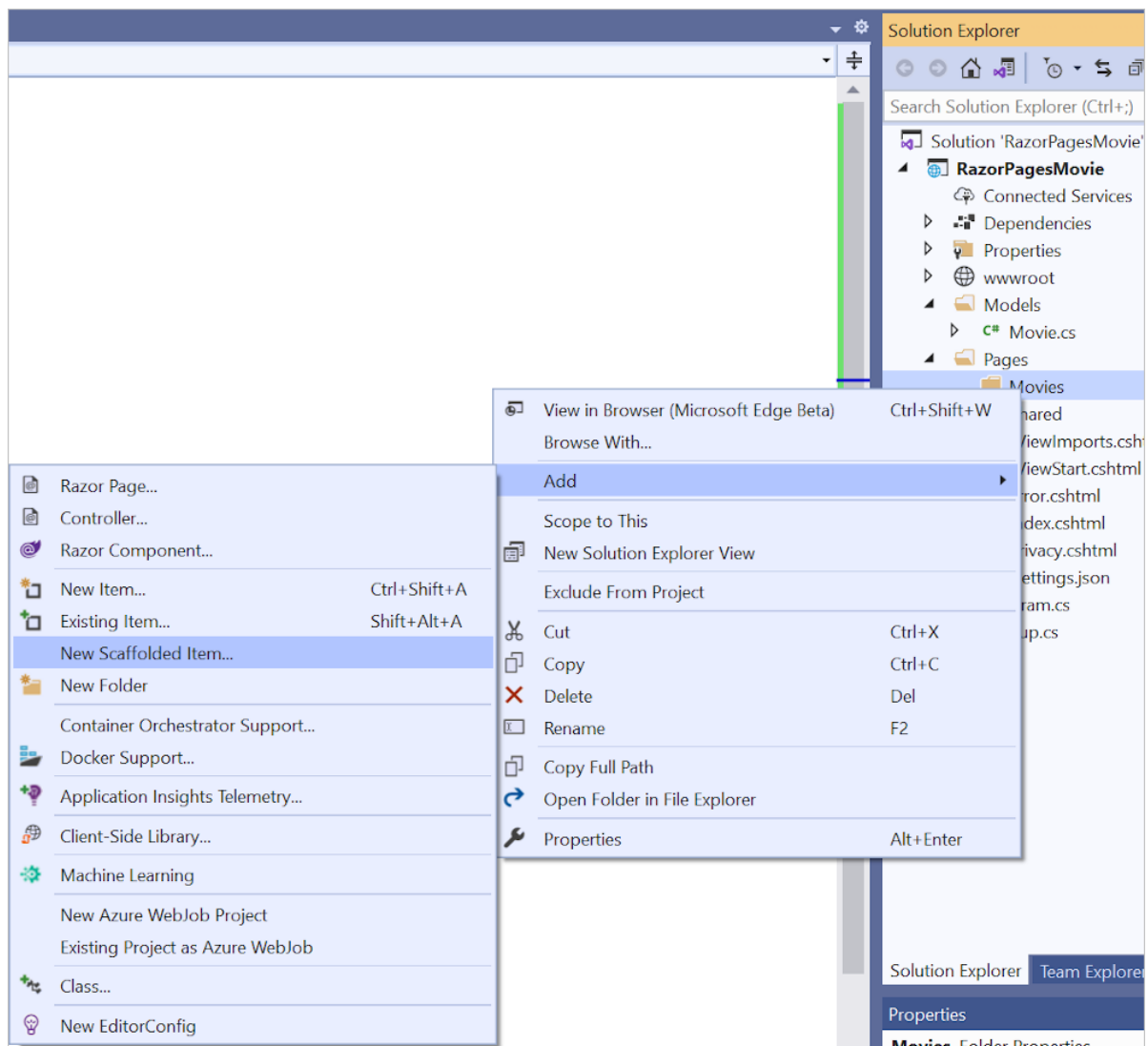
```
using System.ComponentModel.DataAnnotations;
```

```
namespace RazorPagesMovie.Models;
```

```
public class Movie
{
    public int Id { get; set; }
    public string? Title { get; set; }
    [DataType(DataType.Date)]
    public DateTime ReleaseDate { get; set; }
    public string? Genre { get; set; }
    public decimal Price { get; set; }
}
```

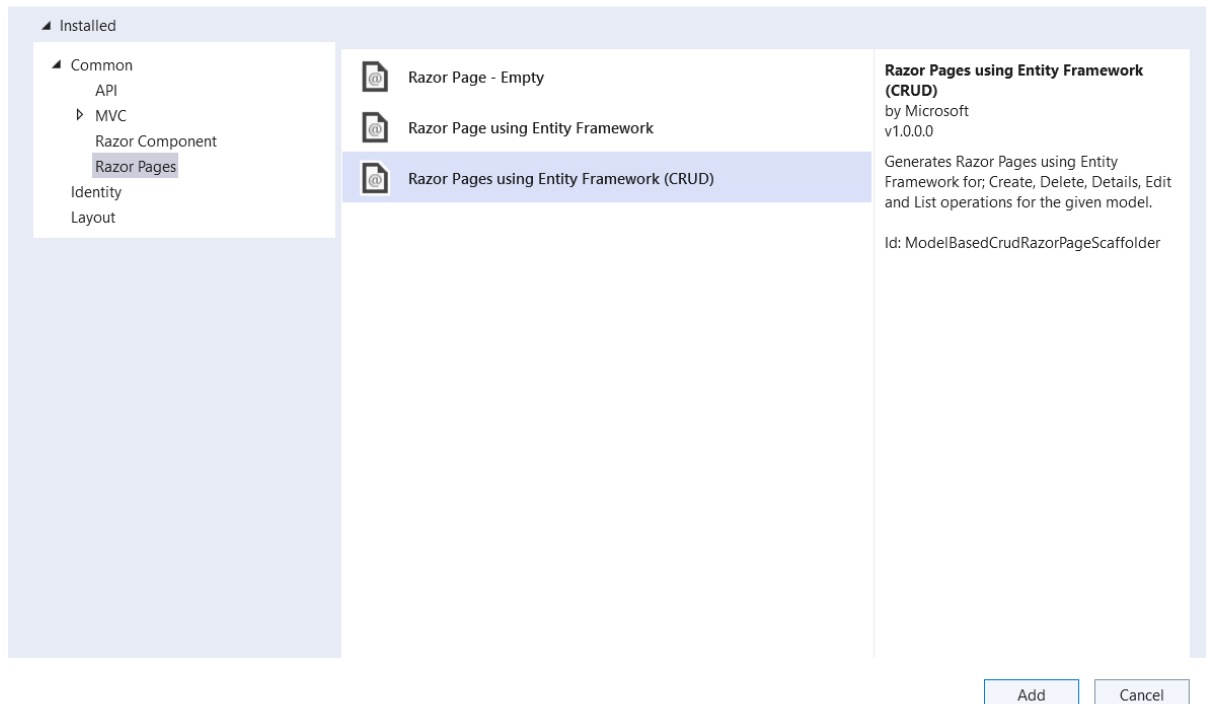
## Scaffold the movie model

1. Create the *Pages/Movies* folder:
  - a. Right-click on the *Pages* folder > **Add** > **New Folder**.
  - b. Name the folder *Movies*.
2. Right-click on the *Pages/Movies* folder > **Add** > **New Scaffolded Item**.



In the **Add New Scaffold** dialog, select **Razor Pages using Entity Framework (CRUD) > Add**.

## Add New Scaffolded Item



Complete the **Add Razor Pages using Entity Framework (CRUD)** dialog:

- In the **Model class** drop down, select **Movie (RazorPagesMovie.Models)**.
- In the **Data context class** row, select the + (plus) sign.
  - In the **Add Data Context** dialog, the class name `RazorPagesMovie.Data.RazorPagesMovieContext` is generated.
- Select **Add**.

Add Razor Pages using Entity Framework (CRUD)

Generates Razor Pages using Entity Framework for; Create, Delete, Details, Edit and List operations for the selected model.

Model class:

Movie (RazorPagesMovie.Models)

Data context class:

RazorPagesMovie.Data.RazorPagesMovieContext

+

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

...

(Leave empty if it is set in a Razor \_viewstart file)

Add

Cancel

Files created and updated

The scaffold process creates the following files:

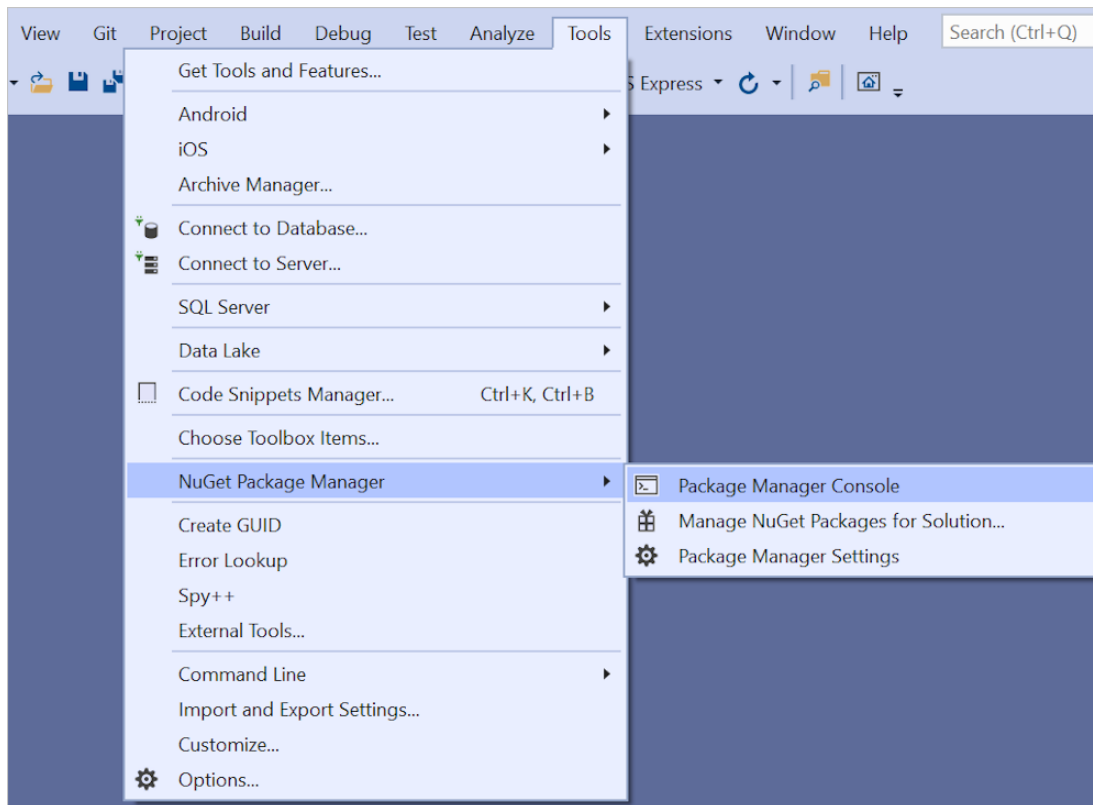
- *Pages/Movies*: Create, Delete, Details, Edit, and Index.
- *Data/RazorPagesMovieContext.cs*

Create the initial database schema using EF's migration feature

In this section, the **Package Manager Console** (PMC) window is used to:

- Add an initial migration.
  - Update the database with the initial migration.
1. From the **Tools** menu, select **NuGet Package Manager > Package Manager Console**.





In the PMC, enter the following commands:

```
Add-Migration InitialCreate  
Update-Database
```

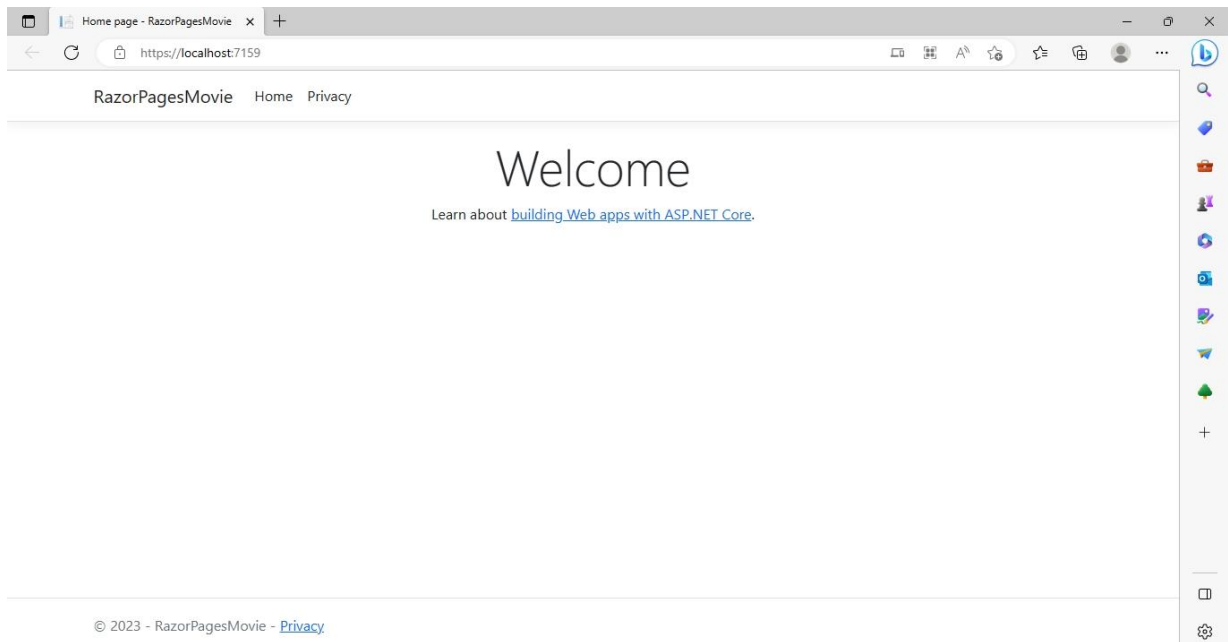
Test the app

1. Run the app and append `/Movies` to the URL in the browser (`http://localhost:port/movies`).

Following will appear in the address bar

<https://localhost:7159/>

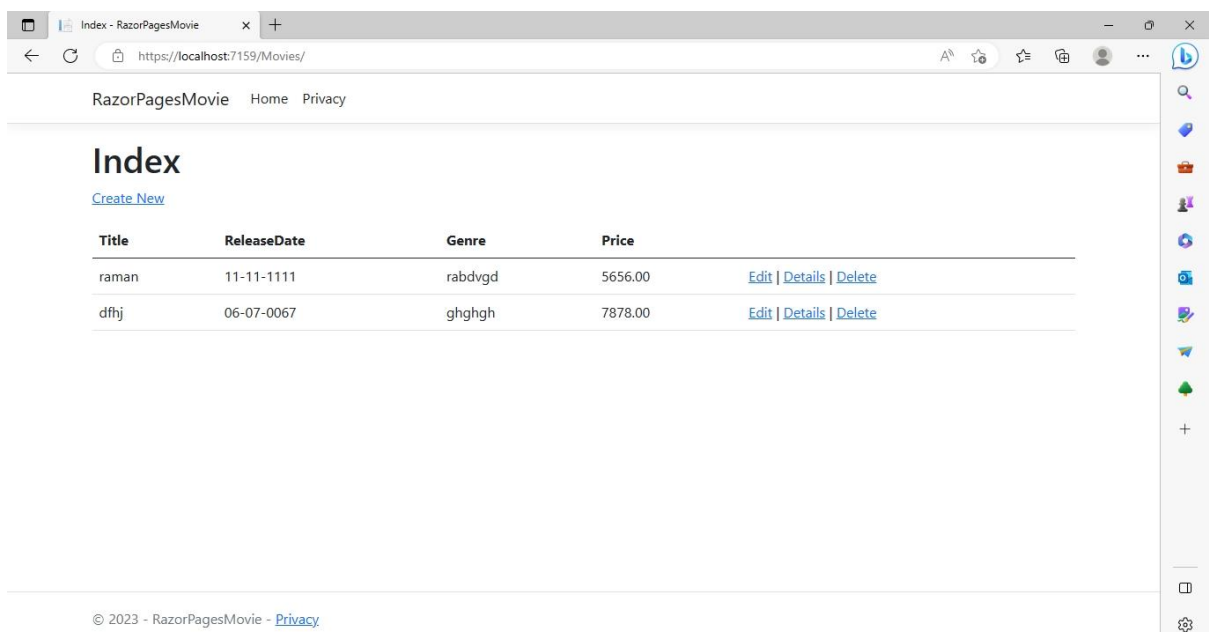
and following will be the output



To View the CRUD Application type

<https://localhost:7159/Movies/>

Following will be the output



Project is Done