Realiza las siguientes modificaciones, que completan el ciclo de una aplicación MVC, en tu proyecto:

1.- Desglosa una **URL**en todos sus secciones. Pon un ejemplo.

https://www.techtarget.com/searchnetworking/search/query?q=URL

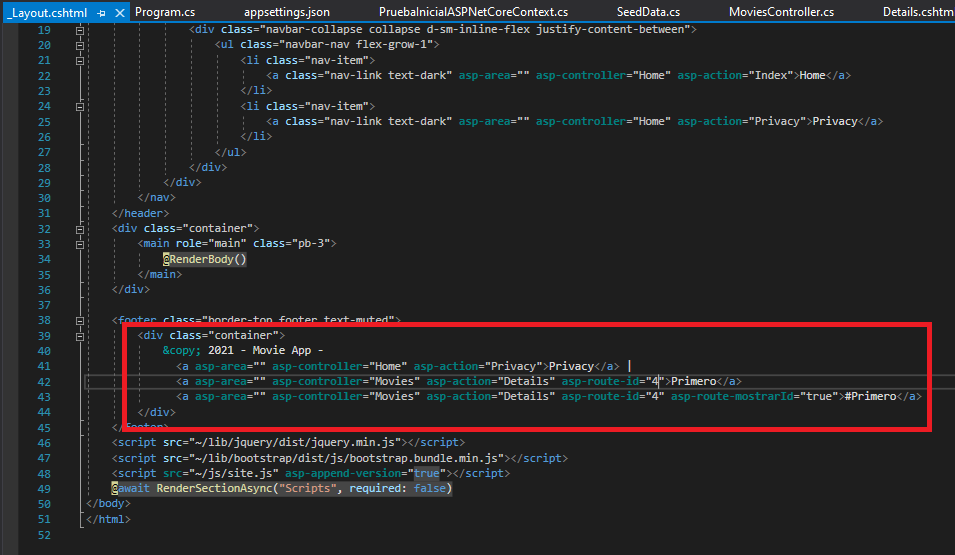
protocolo://subdominio.dominio/ruta/nomberArchivo/query

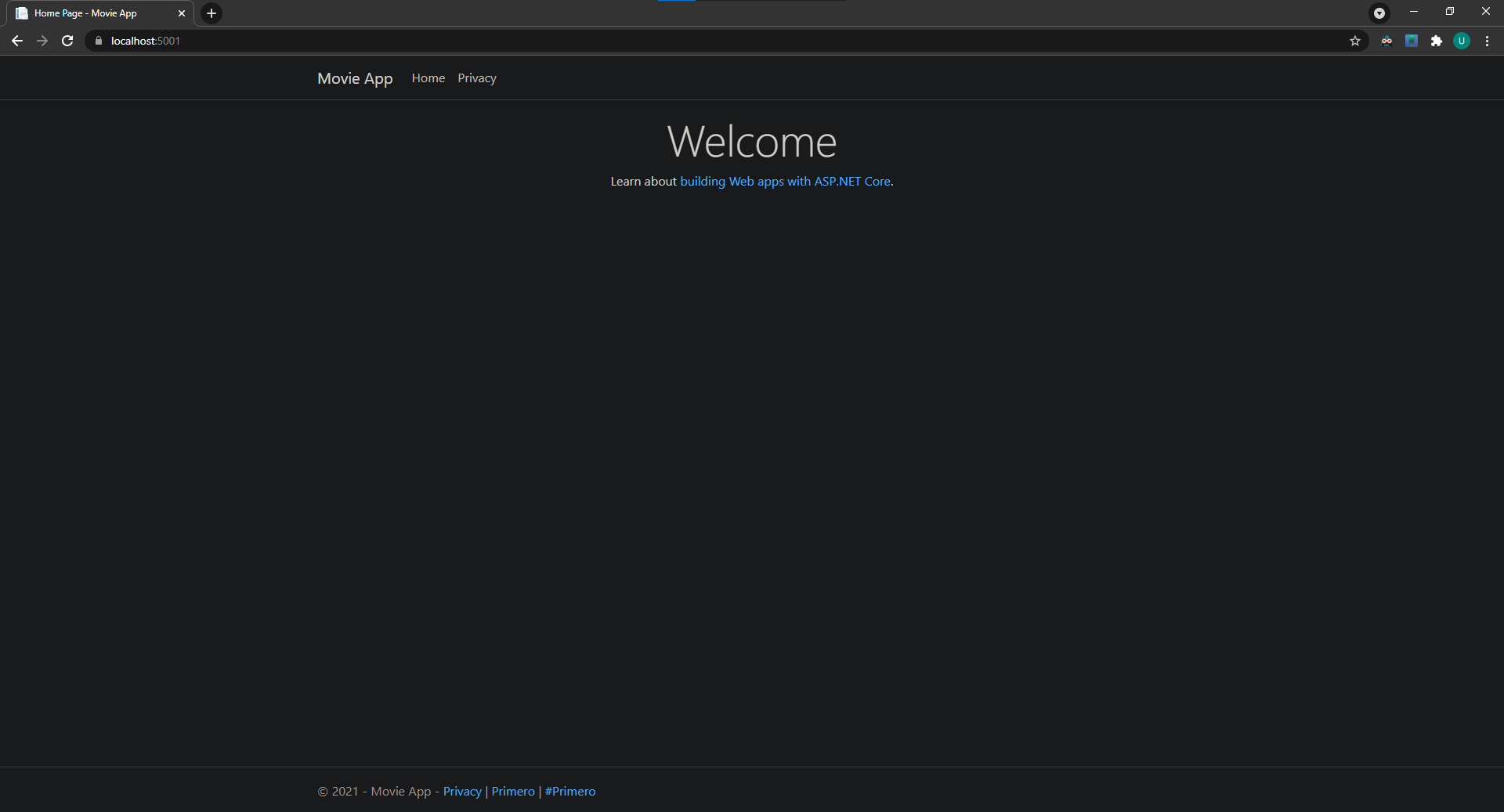
2.- **Escribe**una URL que redirija al controlador y a la acción de Details. Añade un parámetro en la Query de tipo booleano.

<https://localhost:5001/Movies/Details/4?mostrarId=true>   
     Modifica, si lo necesistas, el método adecuado. Esta URL la debes lanzar desde el navegador.

3.- Crea **dos** enlaces en el pie de la página principal.

Los añadimos en esta parte, y al ejecutar el proyecto nos aparecerán los enlaces en su sitio correctamente.



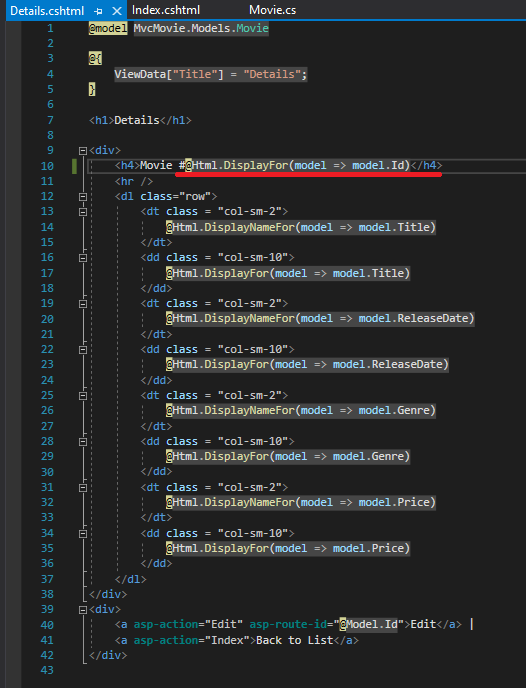


4.- Añade un campo/parámetro **booleano** en la **url**. (idem que el paso anterior)

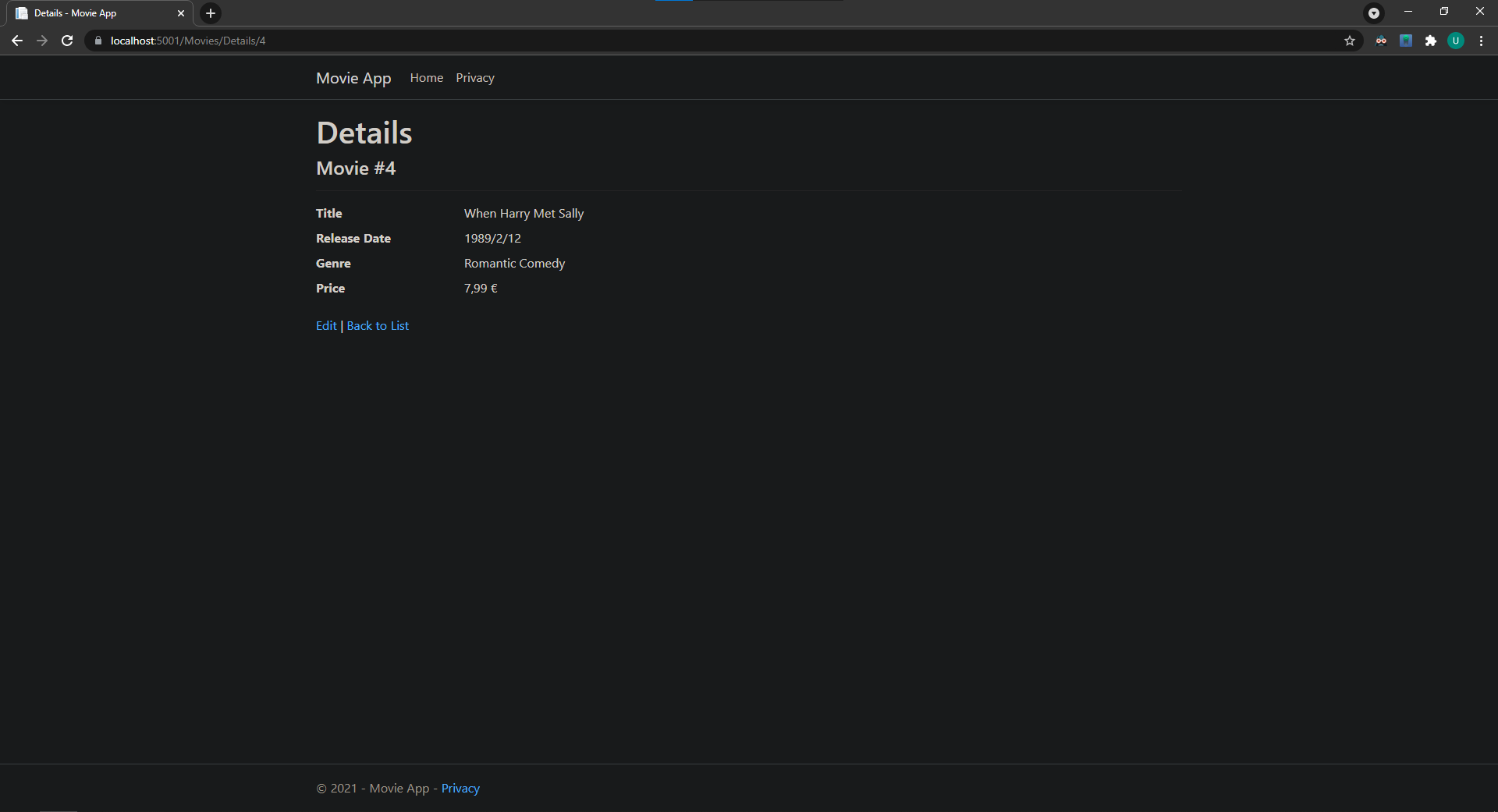
<https://localhost:5001/Movies/Details/5?mostrarId=true>

5.- Modifica en la página de Details, para que aparezca un enlace al **Prev**(previo) y otro al **Next**(siguiente) registro en la tabla Movie. Y también en el titulo **<h4>** de la página. "**Movie #Id**".

Cambiamos el Movies/Details.cshtml para añadir lo siguiente

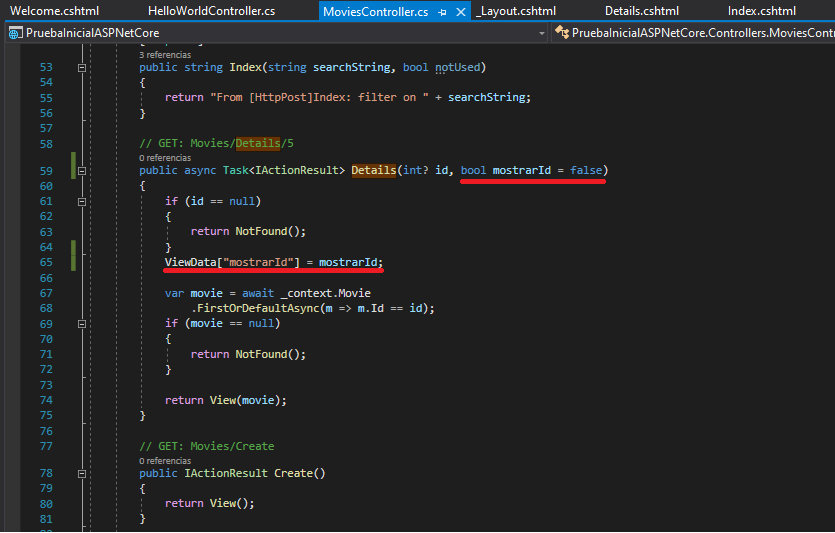


Así ya muestra el id de la película:

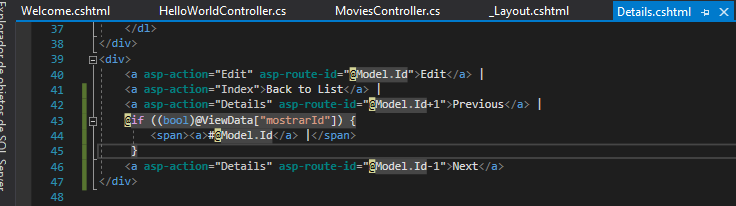


Al pasar el parámetro “mostrarId” como true, se mostrará correctamente.

Para eso, modificamos el código en el Controllers/MoviesController.cs

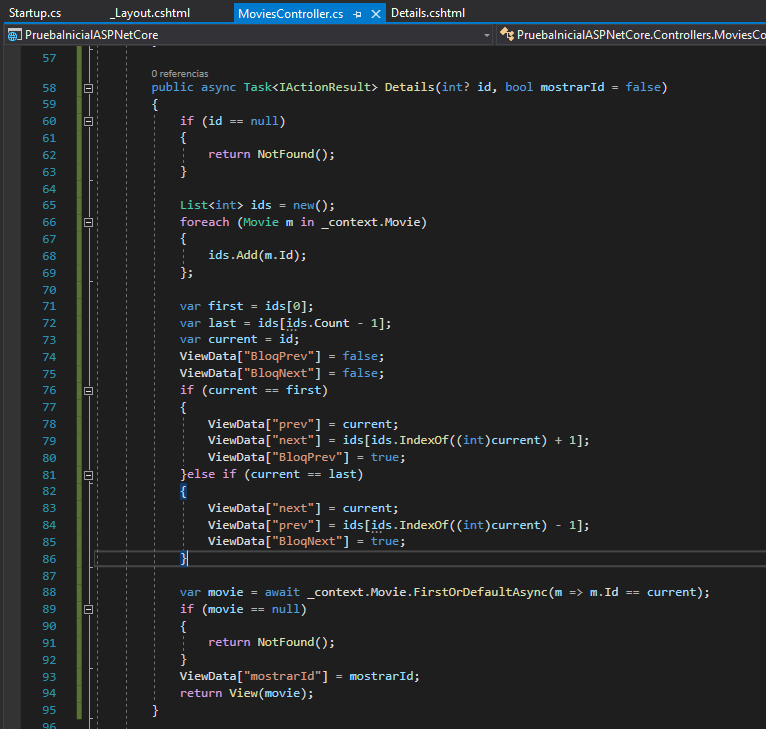


Y en el details.cshtml (los botones previous y next aún no funcionan correctamente.

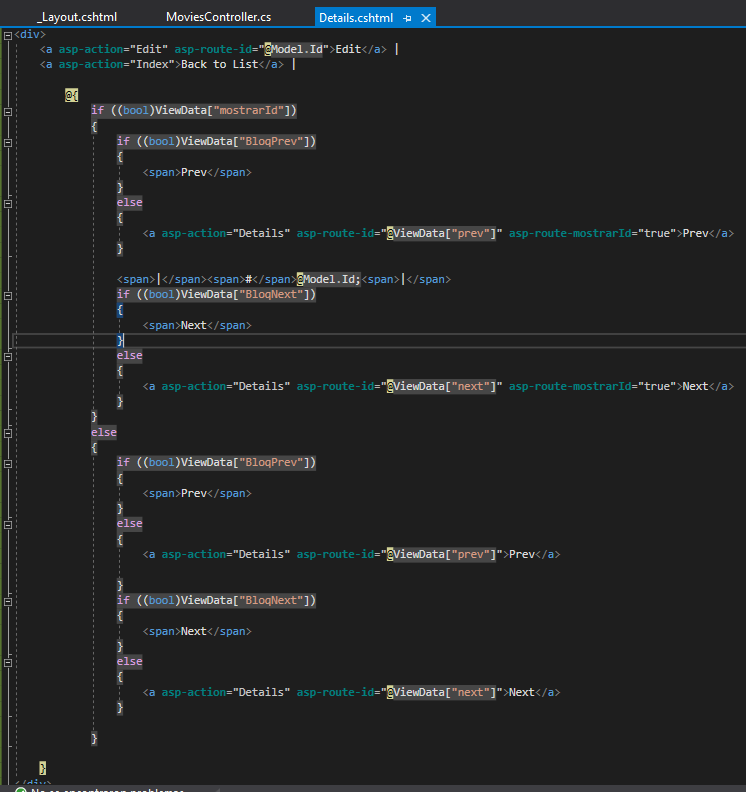


Observa que en Details puede**aparecer** (o no) el**#Id**del Movie actual, en función de ese **parámetro booleano** de la URL, que lo utilizarás para mostrar u ocultarlo.  
  
Al pinchar en cada uno de los enlaces nos llevará al Movie indicado. Si no existe el anterior o el siguiente no se permitirá pulsar el enlace. Si el enlace **Primero#** indica que se muestra el Id, los siguientes Prev/Next lo mostrarán. El enlace **Primero**indica que no se muestra, y no se mostrará el Id en la página.

Código en MoviesController para comprobar si los botones tienen que estar activados o no



Código en Details.cshtml para mostrar los botones activados/desactivados, así como para hacerlos funcionar.



6.- Indica que pasos habría que hacer si en lugar de un parámetro en la **Query** de la URL, queremos escribir una nueva ruta distinta de la anterior, que nos redirija a otro controlador, y todos los parámetros estuviesen escritos en el **Path**de la URL

Tendríamos que ir al Startup.cs, y modificar el siguiente código: donde antes ponía {controller=Home}, ahora hemos puesto Movies

