

Programación 2

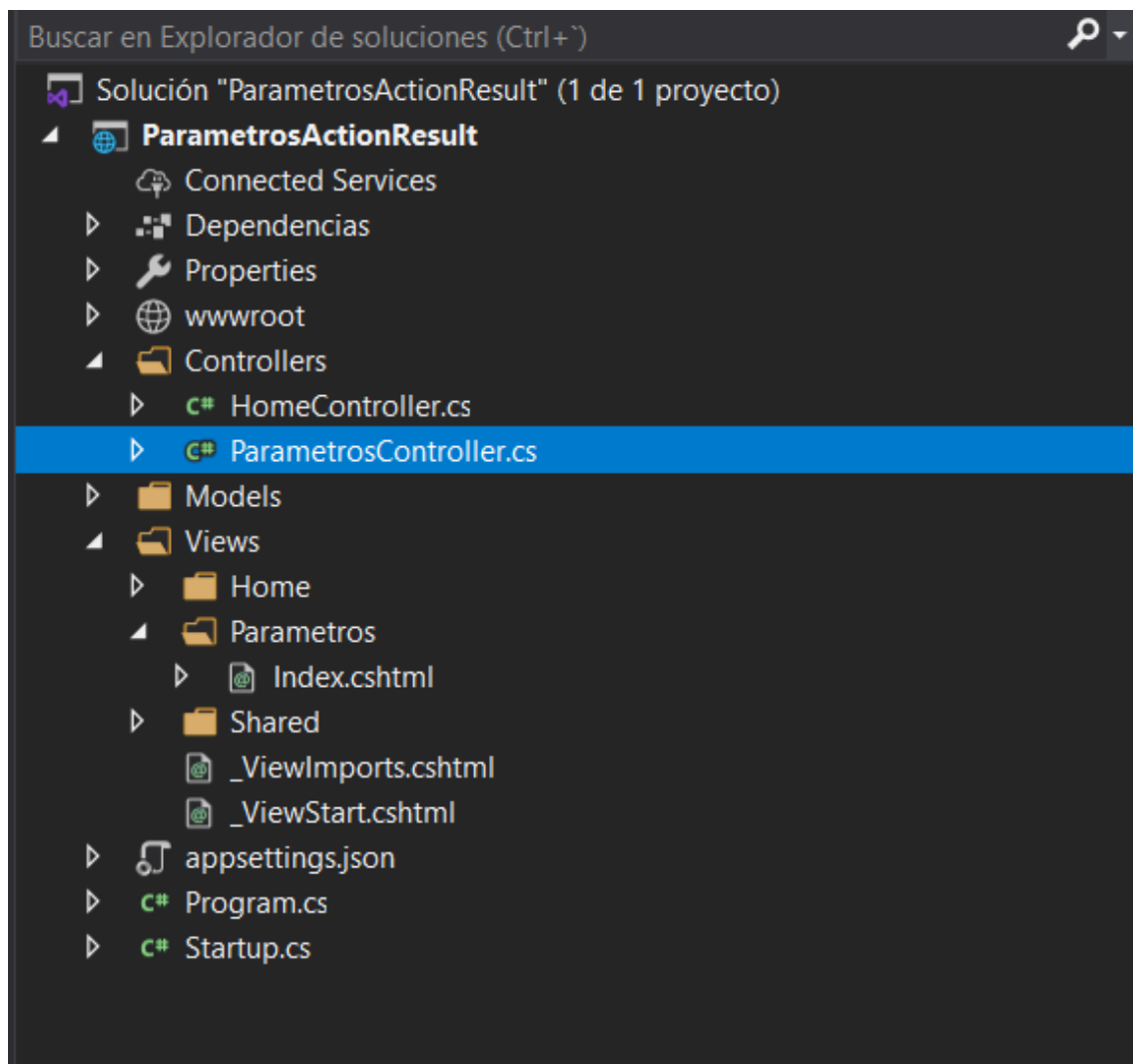
Guía MVC 3 – Métodos de acción

En esta guía veremos cómo se reciben en un método de controlador los valores enviados a través del método HTTP GET, desde una URL escrita en la barra de direcciones del navegador, desde un enlace, o desde un Formulario que tenga el atributo METHOD =GET (o no tenga METHOD seteado, ya que por defecto para HTTP GET).

Para probar es conveniente poner un punto de parada en cada método de acción, así podrá inspeccionar las variables y hacer el seguimiento del flujo de ejecución.

1. Crear un proyecto Web ASP NET Core y nombrarlo "ParametrosActionResult"
2. Crear un controlador llamado "ParametrosController"
3. Crear una carpeta en Views de nombre "Parametros" y dentro de ella una vista Index.cshtml

En este punto debemos tener la siguiente estructura:



Dejaremos el método action result Index que se presenta por defecto en el controller.

Agregar en ParametrosController el siguiente método:

```
0 referencias
public IActionResult ProbarParametro(int id) {
    ViewBag.Mensaje01 = "Recibido el valor" + id.ToString();
    return View("Index");
}
```

En la vista Index.cshtml escribiremos el siguiente código para llamar al método y pasarle el parámetro

```
<h1>Index</h1>

<h2>Index de Parámetros</h2>
<a href="/parametros/ProbarParametro/54">Probar</a>
@ViewBag.Mensaje01
```

Ejecutar la aplicación y acceder al controller y a la vista index para poder probarlo.

Se debe escribir a mano en el navegador la url de la aplicación seguida de /parámetros/Index

Index

Index de Parámetros

[Probar](#) Recibido el valor54

Parte 2.

Realizar el siguiente cambio en la vista:

```
<h1>Index</h1>

<h2>Index de Parámetros</h2>
<a href = "/parametros/ProbarParametro?Id=877">Probar</a>
@ViewBag.Mensaje01
```

En la url escribimos el nombre del parámetro que se debe recibir.

Probar la aplicación nuevamente sin realizar más cambios.

Probar la aplicación nuevamente, pero cambiando el nombre del parámetro del método IActionResult de esta manera:

```
public IActionResult ProbarParametro(int edad) {
    ViewBag.Mensaje01 = "Recibido el valor" + edad.ToString();
    return View("Index");
}
```

Parte 3

Realizar un método en el controller de la siguiente manera.

```
public IActionResult ProbarParametroConOtroNombre(int edad)
{
    if (edad != null)
        ViewBag.Mensaje02 = "Recibido el valor" + edad.ToString();
    else
        ViewBag.Mensaje02 = "No se recibió ningún valor"; return View("Index");
}
```

Y en la vista escribiremos el siguiente código.

```
<p>
  <a href="/parametros/ProbarParametroConOtroNombre?edad=98">
    Probar pasar un valor por la URL a un parámetro que no se llame Id (x ej. edad)</a>
</p>
```


El nombre declarado en el parámetro debe ser el mismo que viaja en la url.

EVITAR QUE SE PRODUZCA UNA EXCEPCIÓN CUANDO UN PARÁMETRO RECIBIDO SEA NULL

Si reciben un valor null en un método de acción, se producirá una excepción. Para evitar esto, existen los tipos nullable.

Se declaran incluyendo el operador "?" y podremos preguntar si son null antes de usarlos, evitando excepciones.

Para probarlo modificar el código del método ProbarParametroConOtroNombre volver a probar el enlace que provocaba la excepción.



```
public IActionResult ProbarParametroConOtroNombre(int? edad)
{
    if (edad != null)
        ViewBag.Mensaje02 = "Recibido el valor" + edad.ToString();
    else
        ViewBag.Mensaje02 = "No se recibió ningún valor"; return View("Index");
}
```