

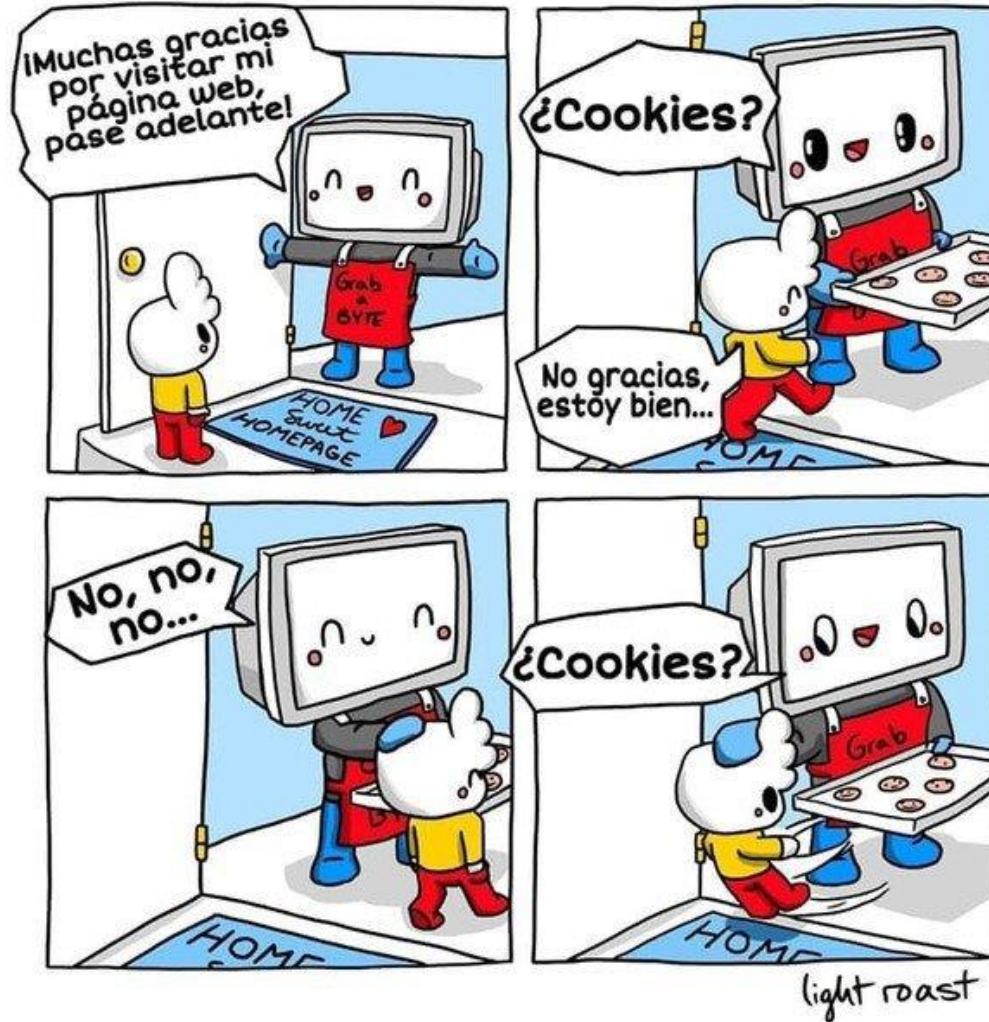
Manejos de sesión

- Cookies.
- Sesiones.
- Configuración.
- Métodos de sesión



Manejos de sesión

Cookies



Manejos de sesión

HTTP

HTTP, el protocolo que se encarga de la comunicación entre un servidor y un cliente en la web, se conoce como protocolo sin estado. En otras palabras, si un usuario solicita dos páginas en un servidor, no se compartirá automáticamente ninguna información entre estas dos solicitudes. En su lugar, un desarrollador tendrá que confiar en algo llamado cookies para compartir información entre las solicitudes (requests). Esto es extremadamente útil en muchas situaciones, por ejemplo, para mantener un usuario conectado entre varias solicitudes, etc

Manejos de sesión

Cookies

Una cookie es básicamente un archivo físico de texto sin formato almacenado por el cliente (normalmente un navegador), vinculado a un sitio web específico. El cliente permitirá a este sitio web específico leer la información almacenada en este archivo en solicitudes posteriores, básicamente permitiendo que el servidor (o incluso el propio cliente) almacene información para su uso posterior.



Manejos de sesión

Sesión

El estado de sesión es un recurso de ASP.NET Core para el almacenamiento de datos de usuario mientras el usuario examina una aplicación web. El estado de sesión usa un almacén mantenido por la aplicación para conservar los datos en las solicitudes de un cliente. Los datos de sesión están respaldados por una memoria caché y se consideran datos efímeros

Estado de sesión

Para mantener el estado de sesión, ASP.NET Core proporciona una cookie al cliente que contiene un identificador de sesión. El identificador de sesión de la cookie:

- Se envía a la aplicación con cada solicitud.
- Lo usa la aplicación para capturar los datos de la sesión.

Manejos de sesión

Como usar sesiones en nuestro código

1 – Instalar el paquete Sesión desde la consola:

`Install-Package Microsoft.AspNetCore.Session`

1 - Para habilitar el middleware de sesión, Startup debe contener:

- Cualquiera de las cachés de memoria **IDistributedCache**.
- Una llamada a **AddSession** en **Services**
- Una llamada a **App.UseSession**

2- Incluir el paquete a nuestros controladores donde queramos usar sesiones

`Microsoft.AspNetCore.Session`

`Microsoft.AspNetCore.Http`

3- Luego, en nuestro código, tendremos acceso al objeto `HttpContext` donde podremos recuperar la sesión que se está ejecutando.

`HttpContext.Session`

Manejos de sesión

Configurando el servicio de sesión

El orden del middleware es importante. Se debe llamar UseSession después de llamar a UseRouting y antes de la llamada a UseEndpoints.

En el siguiente código se configura una variable de sesión que caduca a los 10 segundos de ser creada.

```
builder.services.AddDistributedMemoryCache();

builder.services.AddSession(options =>
{
    options.IdleTimeout = TimeSpan.FromSeconds(10);
    options.Cookie.HttpOnly = true;
    options.Cookie.IsEssential = true;
});
```

Manejos de sesión

Configurando el middleware de sesión

El orden del middleware es importante. Se debe llamar `UseSession` después de llamar a `UseRouting` y antes de la llamada a `UseEndpoints`.

El orden del middleware es importante. Se debe llamar *UseSession* después de llamar a *UseRouting* y antes de la llamada a `UseEndpoints`.

```
[...]  
app.UseSession();  
[...]
```


Manejos de sesión

Usos – Establecer y obtener valores de sesión

Se tiene acceso al estado de sesión desde la clase Controller de MVC con *HttpContext.Session*.

La implementación *ISession* proporciona varios métodos de extensión para establecer y recuperar valores de cadena y enteros. Los nuevos métodos de extensión se encuentran en el espacio de nombres *Microsoft.AspNetCore.Http* y son:

- `Get(ISession, String)`
- `GetInt32(ISession, String)`
- `GetString(ISession, String)`
- `SetInt32(ISession, String, Int32)`
- `SetString(ISession, String, String)`

Ejemplo de uso en el código

```
HttpContext.Session.SetString("Nombre", name);           // establece una variable de sesión denominada  
                                                         "Nombre" con el valor Juan
```

```
string Nombre = HttpContext.Session.GetString("Nombre"); //recupera el contenido de la variable nombre
```