

# Programación Web .NET Core

Módulo 4

# Trabajar con archivos estáticos

## Trabajar con archivos estáticos

Los archivos **HTML**, **JavaScript**, **CSS** o **archivos de imagen** en una solicitud **HTTP** no tienen ningún procesamiento del lado del servidor. No cambian en tiempo de ejecución, son estáticos.

De forma predeterminada, todos los archivos estáticos de una aplicación web deben estar ubicados en la carpeta raíz web: **wwwroot**.

La aplicación **ASP.NET Core** no tiene configurado el acceso de forma predeterminada.

En las aplicaciones **ASP.NET Core**, los archivos estáticos se pueden permitir navegar en los clientes, por medio del uso del llamado **middlewareUseStaticFiles**.



**Nota:** Veremos en detalle la configuración de los middleware.

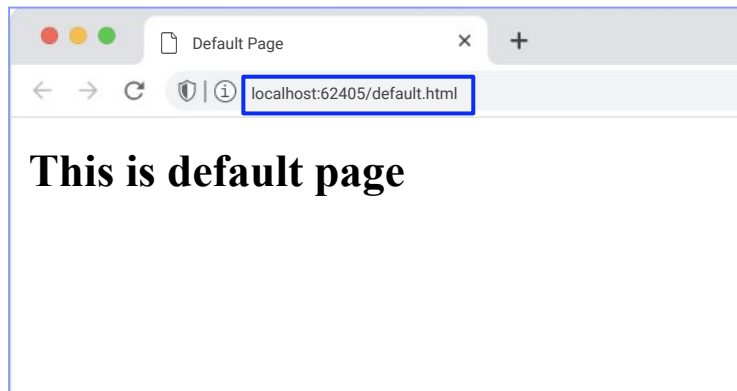
Se deberá agregar el **middlewareStaticFiles** en el método **Configure()** del archivo **startup** como se muestra a continuación:

```
0 referencias
public void Configure(IApplicationBuilder app )
{
    app.UseStaticFiles();
}
```

El método **app.UseStaticFiles()** agrega el **middlewareStaticFiles** a la canalización de solicitudes.

En el navegador, enviar la solicitud:  
**http://localhost:<port>/default.html**

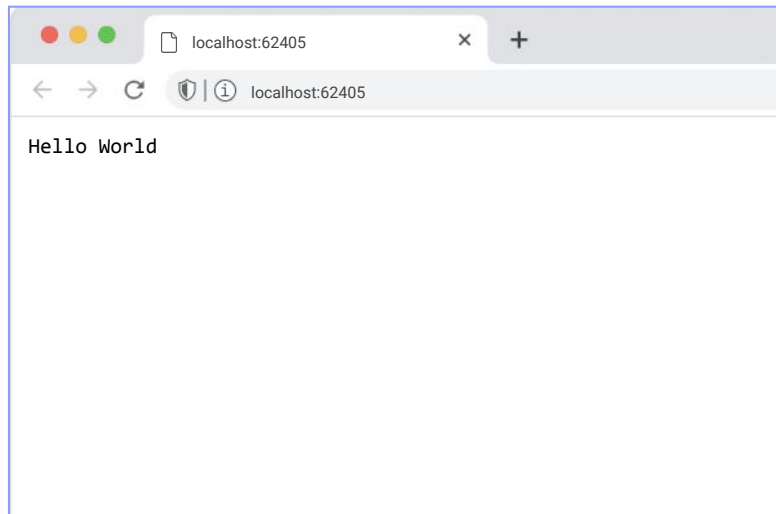
Mostrará **default.html** como respuesta como se muestra en la imagen siguiente:



## Establecer archivo predeterminado

¿Qué pasa si se desea establecer un archivo **html** predeterminado en la solicitud principal?

Actualmente, cuando se envía la solicitud principal **http://localhost:<port>**, se manejará mediante el método de ejecución y mostrará el resultado que vemos en la imagen:



Para configurar **default.html** en la solicitud raíz **http://localhost:<port>**, llamar al método **UseDefaultFiles()** antes **UseStaticFiles()** en el método **Configure** que se muestra aquí:

```
0 referencias
public void Configure(IApplicationBuilder app)
{
    app.UseDefaultFiles();
    app.UseStaticFiles();
}
```

**UseDefaultFiles** configura el llamado **middlewareDefaultFiles** que es una parte de **middleware StaticFiles**. Esto establece automáticamente un archivo **html** llamado **default.html**, **default.htm**, **index.html** o **index.htm** en la solicitud http cuyo nombre es **http://localhost:<port>**.

**Nota:** El orden del middleware es muy importante. **app.UseDefaultFiles()** debe agregarse antes **app.UseStaticFiles()** en la canalización de solicitudes.



## FileServer

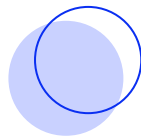
El **middlewareFileServer** combina las funcionalidades del **middlewareUseDefaultFiles** y **UseStaticFiles**.

Entonces, en lugar de usar ambos middleware, simplemente usar **UseFileServer** en el método **Configure**.

```
UseFileServer = UseDefaultFiles + UseStaticFiles
```

Ejemplo de UseFileServer:

```
0 referencias
public void Configure(IApplicationBuilder app)
{
    app.UseFileServer();
}
```



## Configurar archivos estáticos de una carpeta diferente

Es posible configurar **middleware** para acceder a los archivos estáticos de otras carpetas junto con la carpeta raíz web predeterminada en el **wwwroot**. Por ejemplo, la página **admin.html** se agregó a la carpeta de **admin** y también a **test.html** desde la carpeta **wwwroot**.

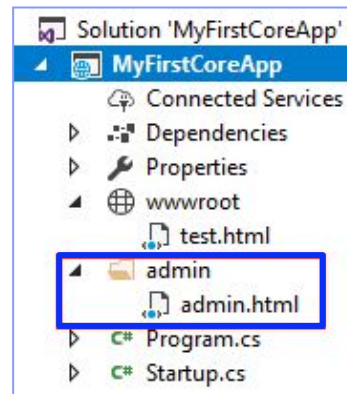
Configurar el **middlewareStaticFiles** en el método **Configure()** de la clase **Startup** como se muestra a continuación. En el código de la imagen se configura el acceso a los archivos estáticos en la carpeta o ruta de solicitud **/admin**.

Veamos el ejemplo en la próxima pantalla.





```
0 referencias
public void Configure(IApplicationBuilder app)
{
    app.UseStaticFiles();
    app.UseStaticFiles(new StaticFileOptions()
    {
        FileProvider = new PhysicalFileProvider(Path.Combine
            (Directory.GetCurrentDirectory(), "Content")),
        RequestPath = new PathString("/Admin")
    });
}
```



Ahora podremos ejecutar la solicitud **HTTP**:  
**http://localhost:1234/admin/admin.html**

**¡Sigamos  
trabajando!**

