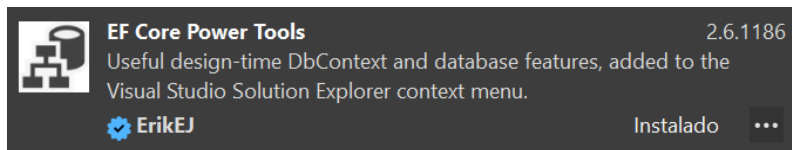


Ingeniería inversa con EF Core PowerTools

Instalamos EF Core Power Tools:

Menú Extensiones -> Administrar extensiones y Examinación:



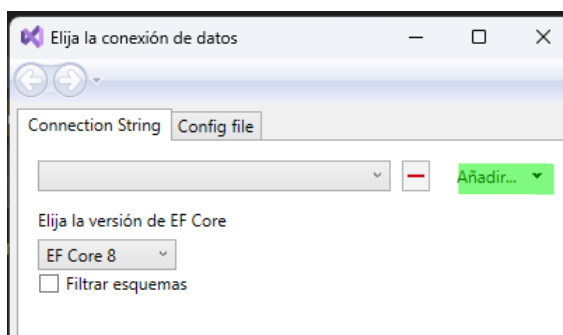
Instalar. Cerrar VS para que se pueda instalar.

Volvemos a abrir VS.

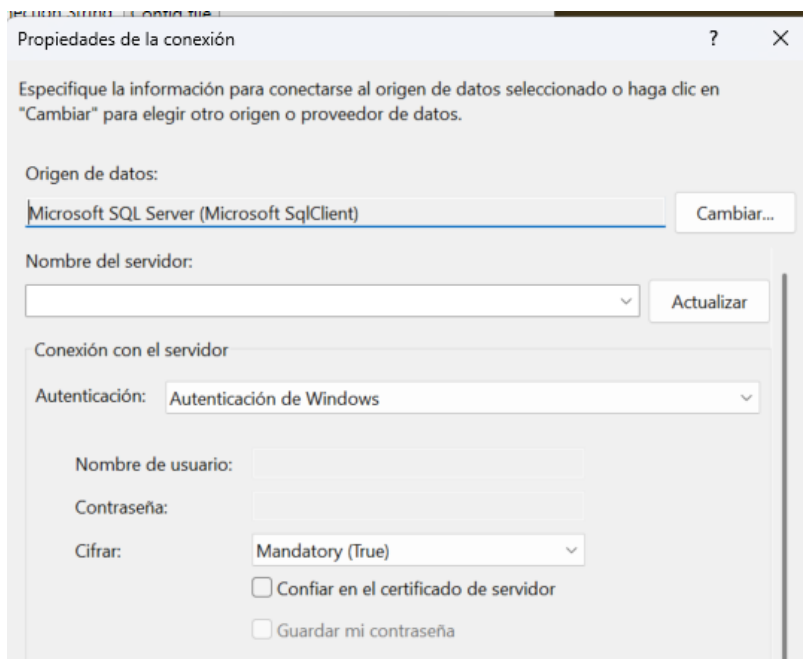
Click derecho en el proyecto y aparece EF Core Power Tools

Seleccionamos ingeniería inversa.

Nos aparece la ventana:



Como no tenemos ninguna cadena de conexión aún, añadimos:



Recordar, cuando instalamos

SQL Server Management Studio

21.6.17

Teníamos estos datos (el nombre del servidor lo guardamos en un documento .txt en el escritorio). Si no lo tenéis, abrir la aplicación.

Conectar con el servidor

SQL Server

Inicio de sesión | Propiedades de conexión | Always Encrypted | Parámetros de conexión a

Servidor

Tipo de servidor: Motor de base de datos

Nombre del servidor: NUPC\SQLEXPRESS

Autenticación: Autenticación de Windows

Nombre de usuario: NuPC\Nuria López Álvarez

Contraseña:

☐ Recordar contraseña

Seguridad de la conexión

Cifrado: Obligatorio

☒ Confiar en el certificado de servidor

Nombre de host en el certificado:

Pues ahora:

Origen de datos:

Microsoft SQL Server (Microsoft SqlClient) Cambiar...

Nombre del servidor: NUPC\SQLEXPRESS Actualizar

Conexión con el servidor

Autenticación: Autenticación de Windows

Nombre de usuario:

Contraseña:

Cifrar: Mandatory (True)

☒ Confiar en el certificado de servidor

☐ Guardar mi contraseña

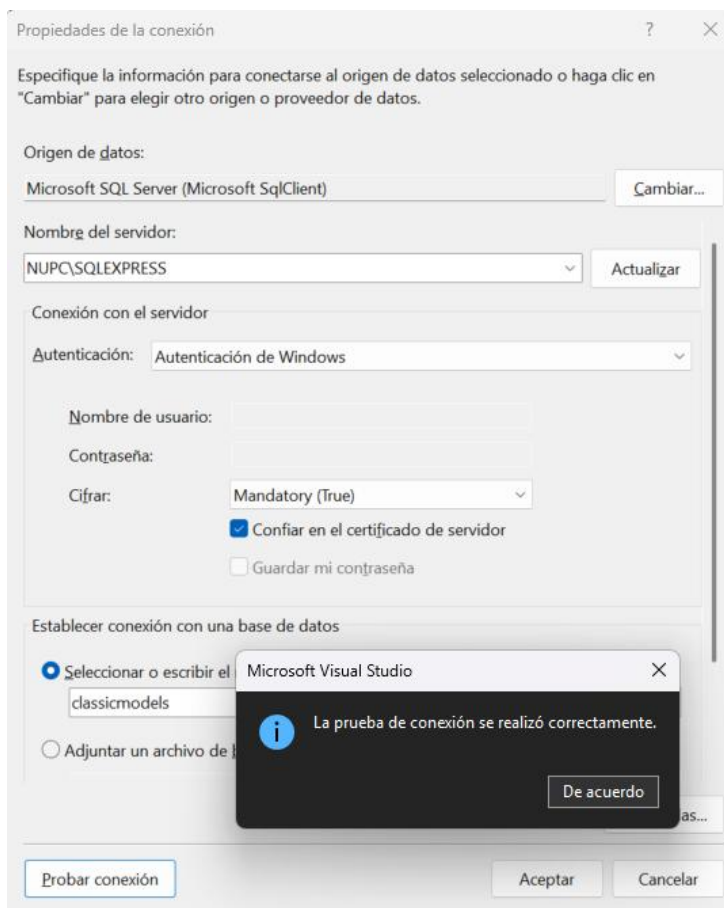
Establecer conexión con una base de datos

☒ Seleccionar o escribir el nombre de la base de datos: classicmodels

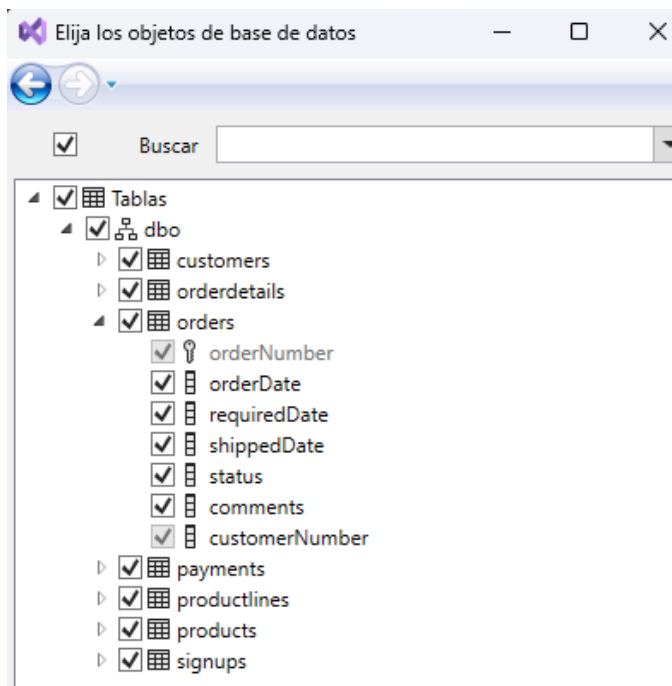
☐ Adjuntar un archivo de base de datos:

Probar conexión Avanzadas... Aceptar Cancelar

Probamos la conexión para asegurarnos que todo es correcto:

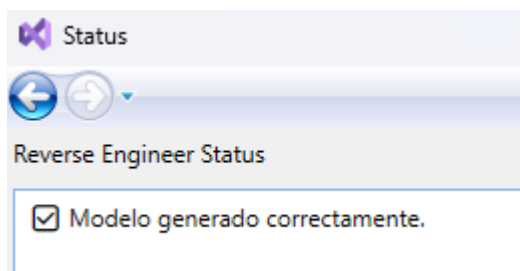
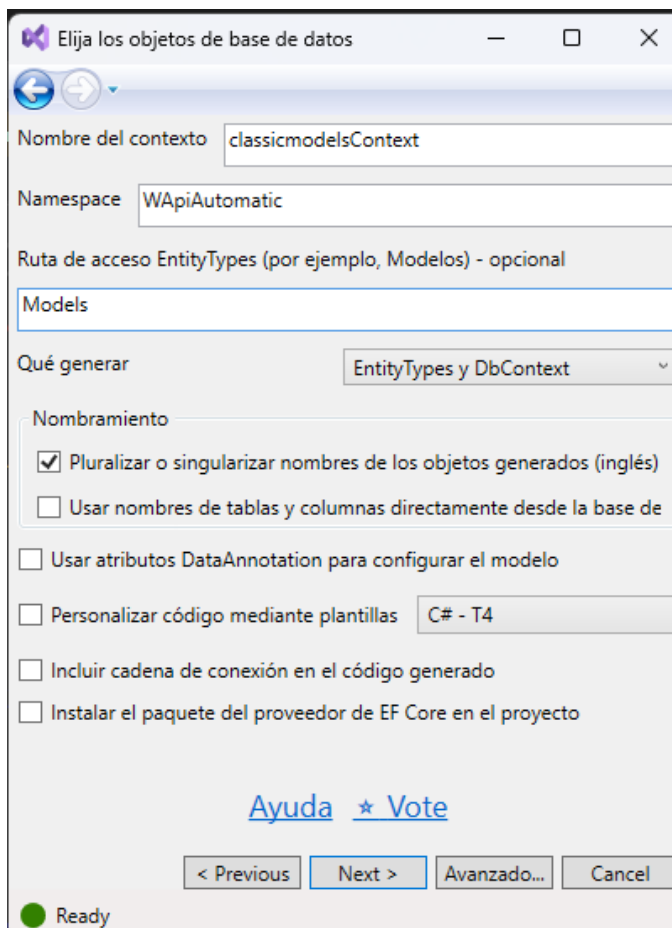


Aceptamos y



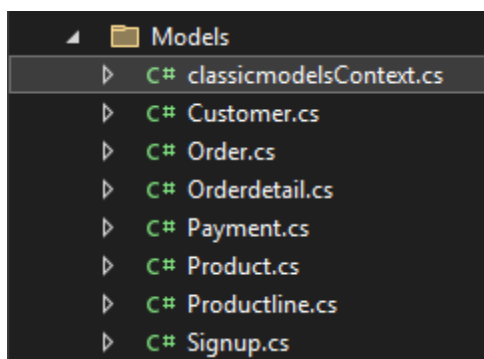
Seleccionamos los datos que nos interesen.

Vamos a seleccionar todas las tablas y Next.



Y listo.

En Modelos tenemos ahora las clases que representan cada elemento de cada tabla.

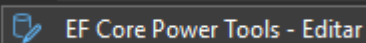


También tenemos el contexto (classicmodelsContext.cs), donde están las entidades y la configuración de la base de datos.

En el caso de que una columna de la tabla se llame como la tabla, en la clase la renombra (le añade 1, por ejemplo) porque una propiedad no se puede llamar como la clase.

Además de crearnos los archivos de c#, también nos crea en archivo json: efpt.config.json. Aquí está almacenada la configuración de la ingeniería inversa. Si queremos cambiar algo, pulsamos botón

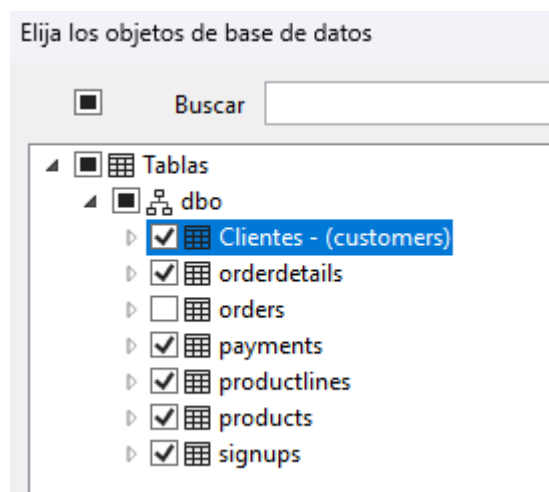
derecho encima y elegimos

A dark button with a light blue icon of a document and a pencil, followed by the text "EF Core Power Tools - Editar".

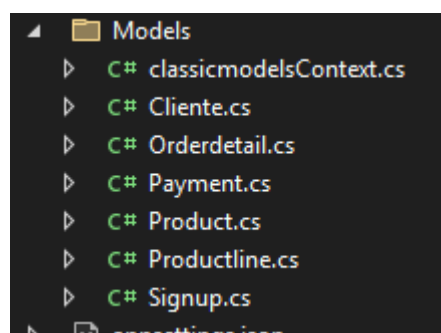
Podremos cambiar los nombres de tablas y/o columnas pulsando F2 sobre el elemento que deseemos.

También podemos quitar o añadir tablas y/o columnas.

Si quitamos la selección de la tabla orders:

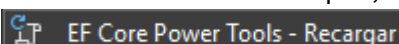


Pulsamos Ok y vemos que desaparece la clase que la representa en Models:



Útil para quitar prefijos, estandarizar los nombres (español, inglés), ...

Si modificamos la base de datos (como añadir tablas o campos, cambiar un tipo de datos), podemos actualizar aquí nuestros modelos

A dark button with a light blue icon of a document and a circular arrow, followed by the text "EF Core Power Tools - Recargar".