

Programa Académico Desarrollo de Software Nivel II

C# II
Prof. Gabriel Roa

Tema II – SQL Server

Contenido

I – SQL SERVER	2
I.1 – INSTALACIÓN	2
I.2 – SQL SERVER MANAGEMENT STUDIO (SSMS)	9
I.3 – CREACIÓN DE BASES DE DATOS	15
I.4 – CREACIÓN DE TABLAS	18
I.4.1 – RELACIONES Y TIPOS DE DATOS	20
I.4.2 – MODIFICACIÓN DE TABLAS	30
I.4.3 – GENERACIÓN DE DIAGRAMA MER	31
I.5 – EJECUCIÓN DE SENTENCIAS	33
I.6 – EJERCICIOS PROPUESTOS	38
II – REFERENCIAS BIBLIOGRÁFICAS	41

I – SQL Server

SQL es un lenguaje de modelado que puede ser implementado bajo diferentes esquemas de manejadores de bases de datos, con la finalidad de crear, ejecutar y administrar una base de datos. En este orden de ideas, cada empresa de desarrollo ha sido libre en su historia de desarrollar su propio manejador de bases de datos, ajustado a sus propias necesidades y características propias.

Microsoft, en 1989, lanza SQL Server, que se convierte en su propio manejador de bases de datos con un sinfín de características que lo convierten en uno de los principales competidores en este mercado. Ofreciendo un aspecto transaccional, con capacidad de ejecución de procedimientos almacenados, un modo de cliente – servidor nativo que facilita la conexión desde dispositivos de la red, un sistema de gestión de credenciales anclado al propio Windows que incrementa su seguridad, SQL Server es actualmente uno de los manejadores de bases de datos más utilizados a nivel empresarial.

En este curso, se hará uso de SQL Server 2012 SP4 Express.

I.1 – Instalación

Para instalar SQL Server 2012, sólo hace falta descargar el instalador desde el aula virtual, y luego, ejecutarlo. Cuando se ejecute, se podrá visualizar una pantalla similar a la siguiente:

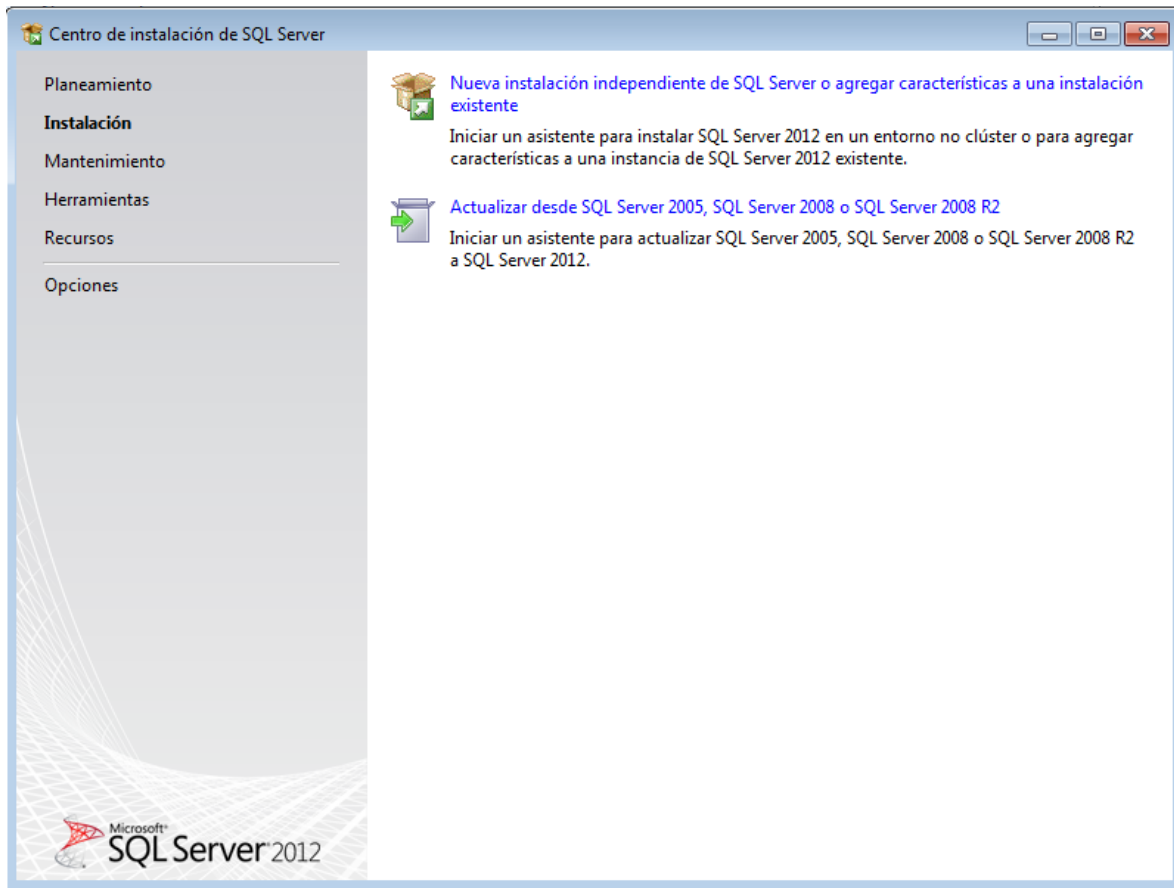


Fig. 66 – Primera ventana previa a la instalación de SQL Server 2012

Debemos hacer click en Nueva Instalación, tras lo cual nos aparecerá una pantalla para aceptar los términos y condiciones. Tras esto, comenzará una preinstalación de los archivos de configuración necesarios para proceder a la instalación:

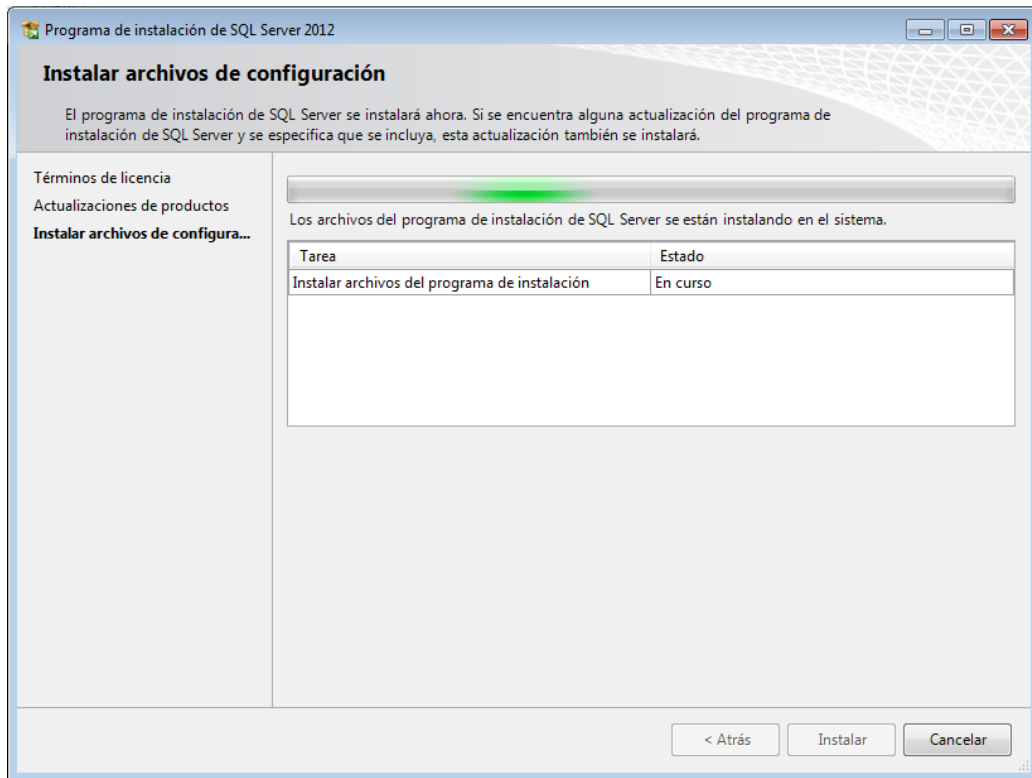


Fig. 67 – Preinstalación en curso

Cuando esta preinstalación finalice, podremos seleccionar las características a instalar:

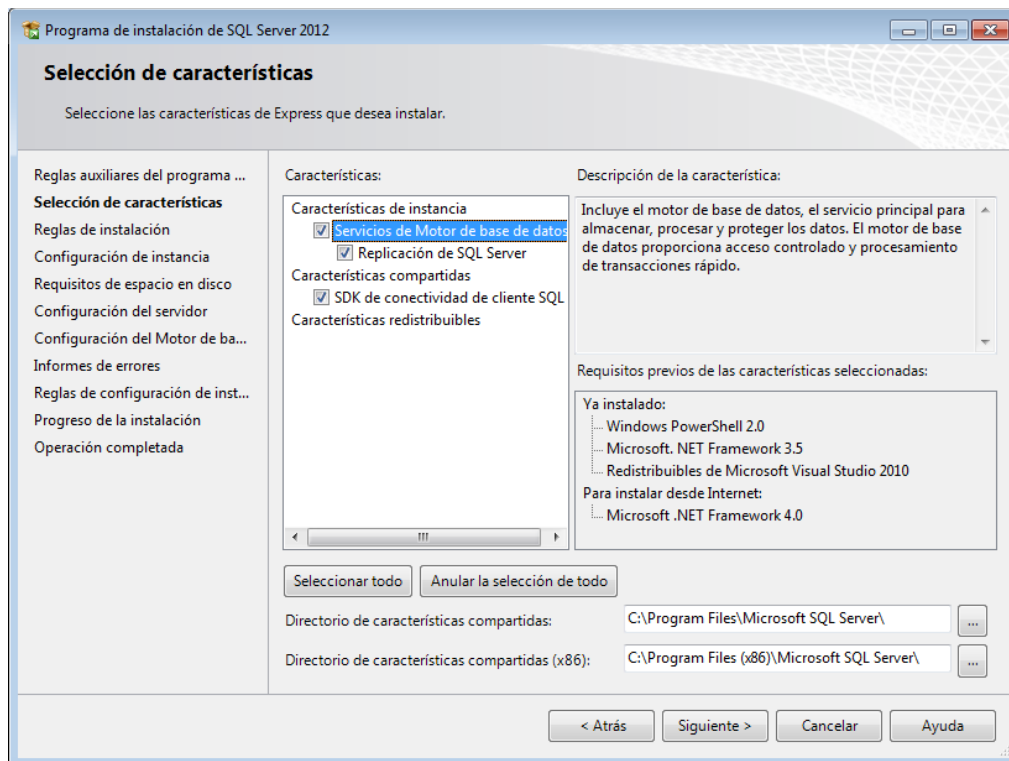


Fig. 68 – Selección de características

Para efectos de este curso, vamos a instalar todas las características que ofrece el instalador.

Ahora, a continuación, podremos crear nuestras instancias. Una instancia en SQL Server consiste en un *Servidor*, un clúster que contiene un conjunto de una o más bases de datos. En este caso, crearemos una instancia llamada CursoCSharp2, cambiando el nombre sólo en el cuadro de texto de Instancia con nombre.

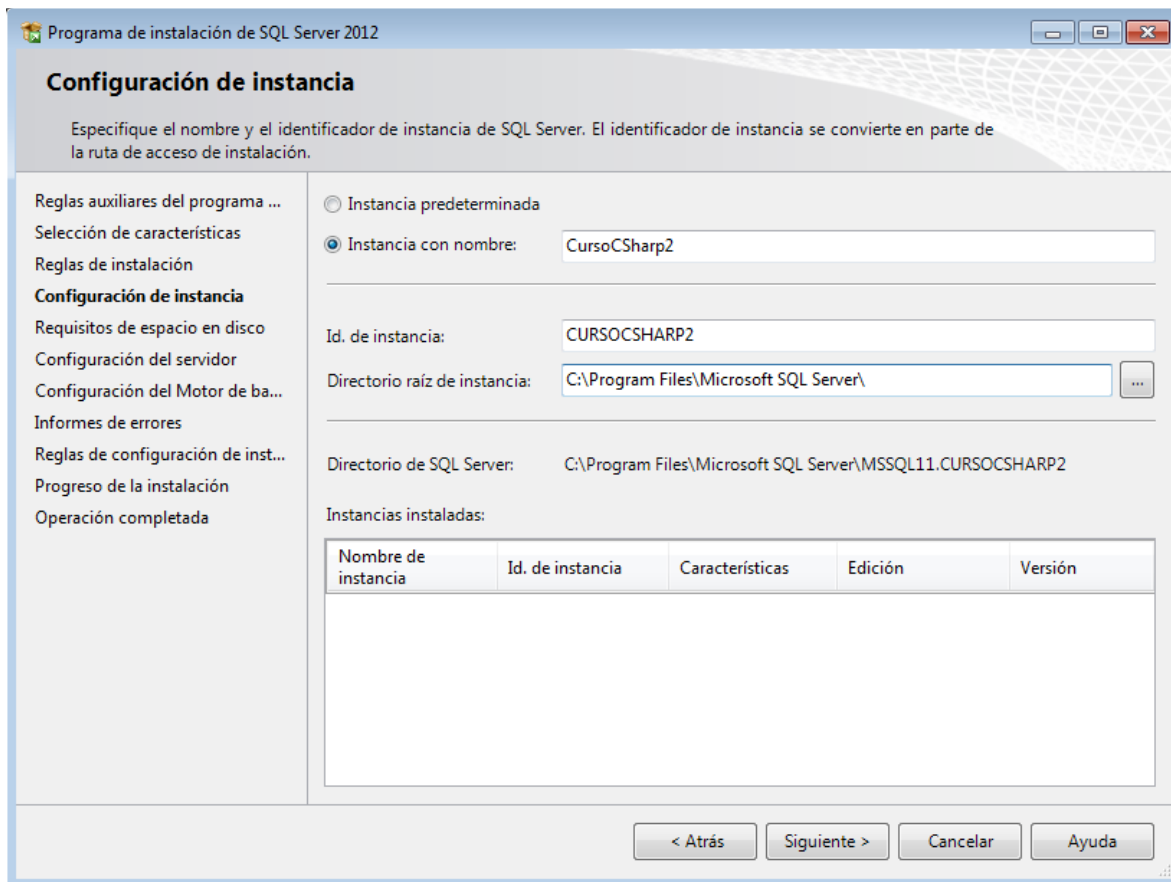


Fig. 69 – Creando una instancia

A continuación, el instalador nos mostrará las cuentas que se crearán en el Sistema Operativo. Tal y como quedó explicado anteriormente, SQL Server puede utilizar un sistema de autenticación basada en cuentas de usuario del propio Windows, por lo que no será necesario establecer un usuario y una clave para poder acceder a las bases de datos.

En este caso, el instalador recomienda la creación automática de dos cuentas, una cuenta por servicio, por lo que hacemos click en Siguiete.

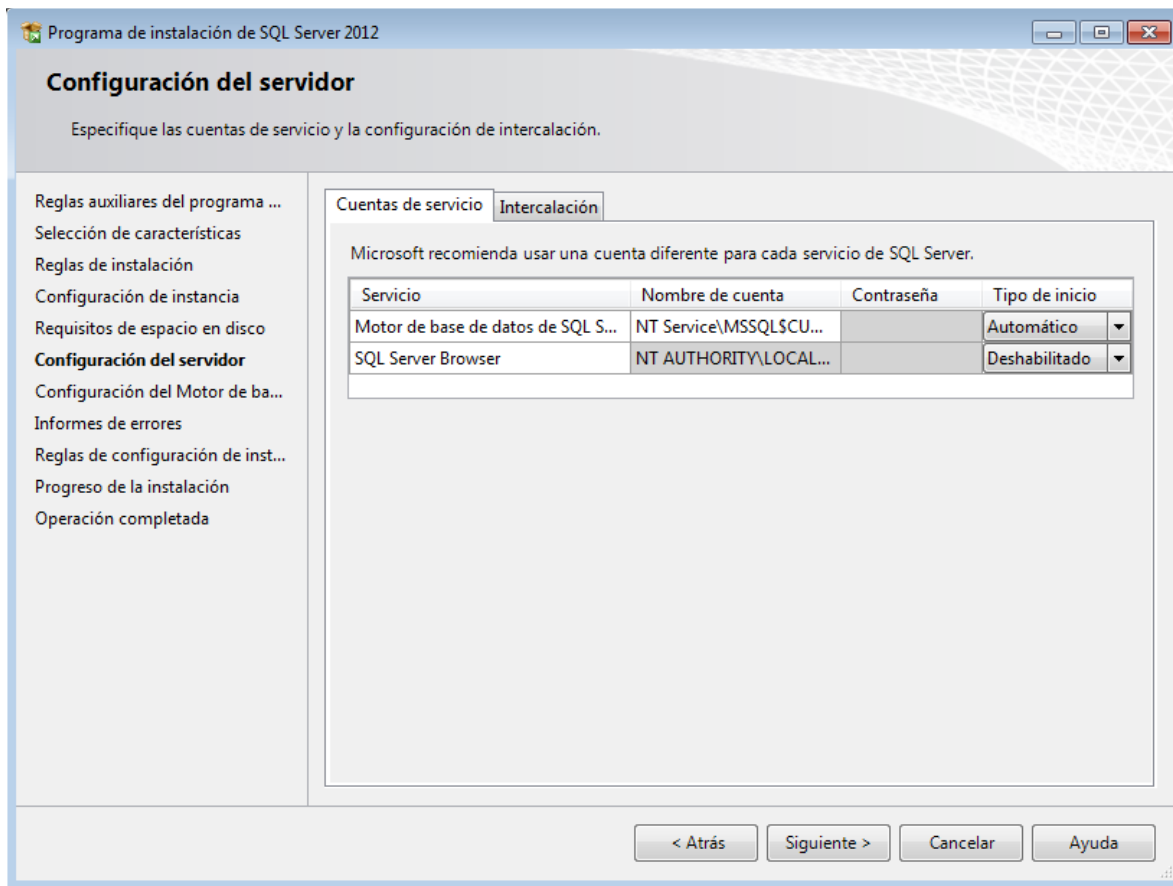


Fig. 70 – Cuentas de usuario a crearse en el SO

Luego, el instalador nos mostrará una pantalla en que podremos configurar el motor de la base de datos. En este, podremos definir el modo de autenticación a seguir – bien sea el de autenticación de Windows o un modo mixto. Haremos uso del modo mixto, por lo que deberemos establecer una contraseña para acceder a esta base de datos, utilizando el superusuario *sa*.

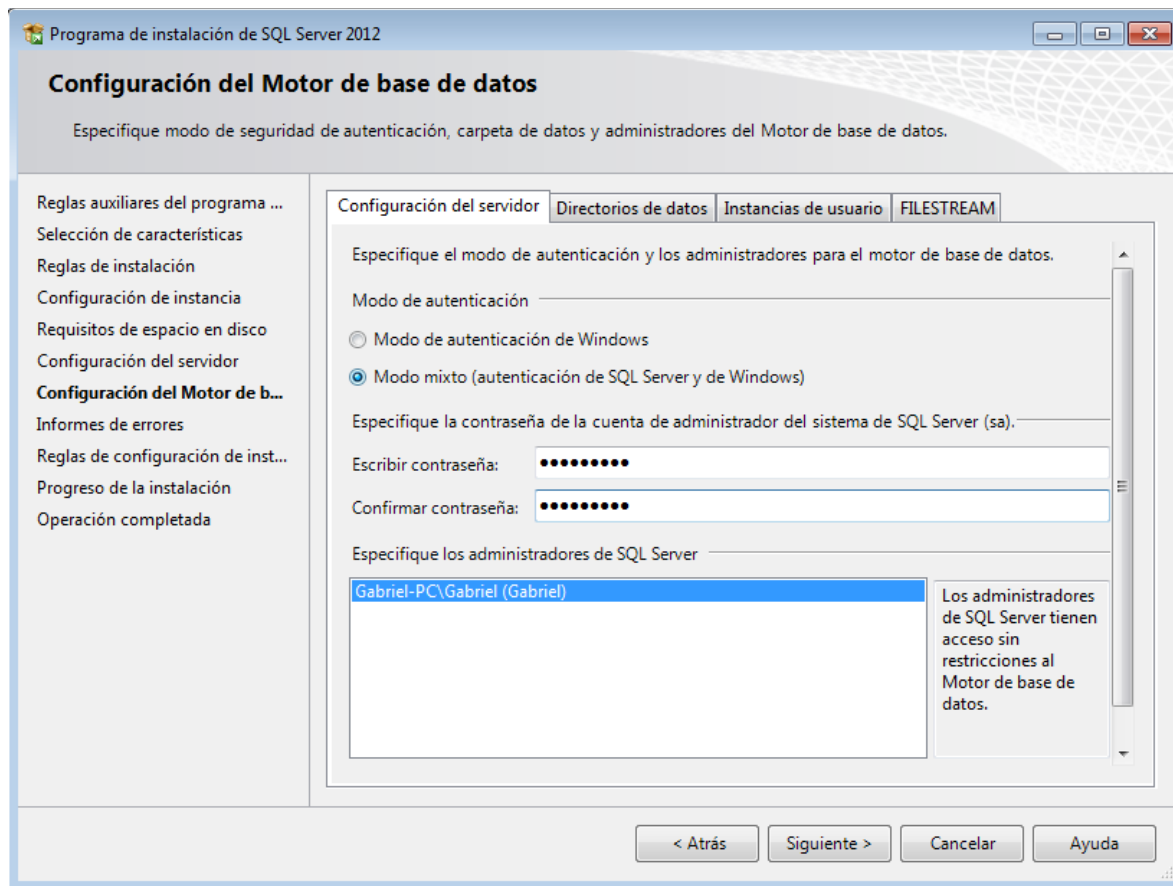


Fig. 71 – Autenticación mixto con SQL Server y Windows

Hacemos click en Siguiente, tras lo cual el instalador nos dará indicaciones de los informes de errores y cómo se manejan. Click en siguiente de nuevo, y la instalación comenzará.

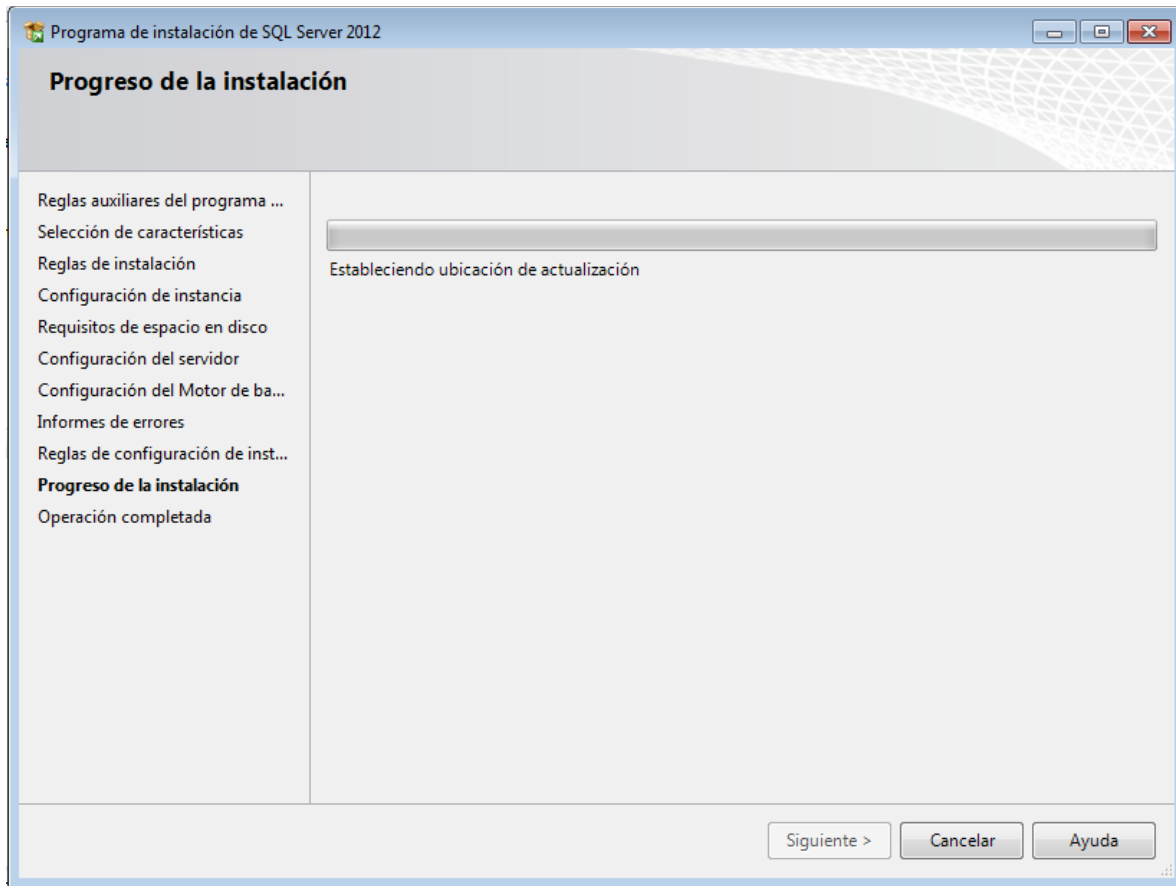


Fig. 72 – Instalación en progreso

I.2 – SQL Server Management Studio (SSMS)

SQL Server cuenta con una herramienta de manejo de las bases de datos llamada SQL Server Management Studio. Este visor nos permite visualizar y crear bases de datos, tablas, registros, además de ejecutar sentencias y configurar y gestionar en general todo lo relacionado con nuestra instancia, incluida la creación de nuevas instancias.

Su instalación es similar a la instalación del SQL Server, a través de otro instalador.

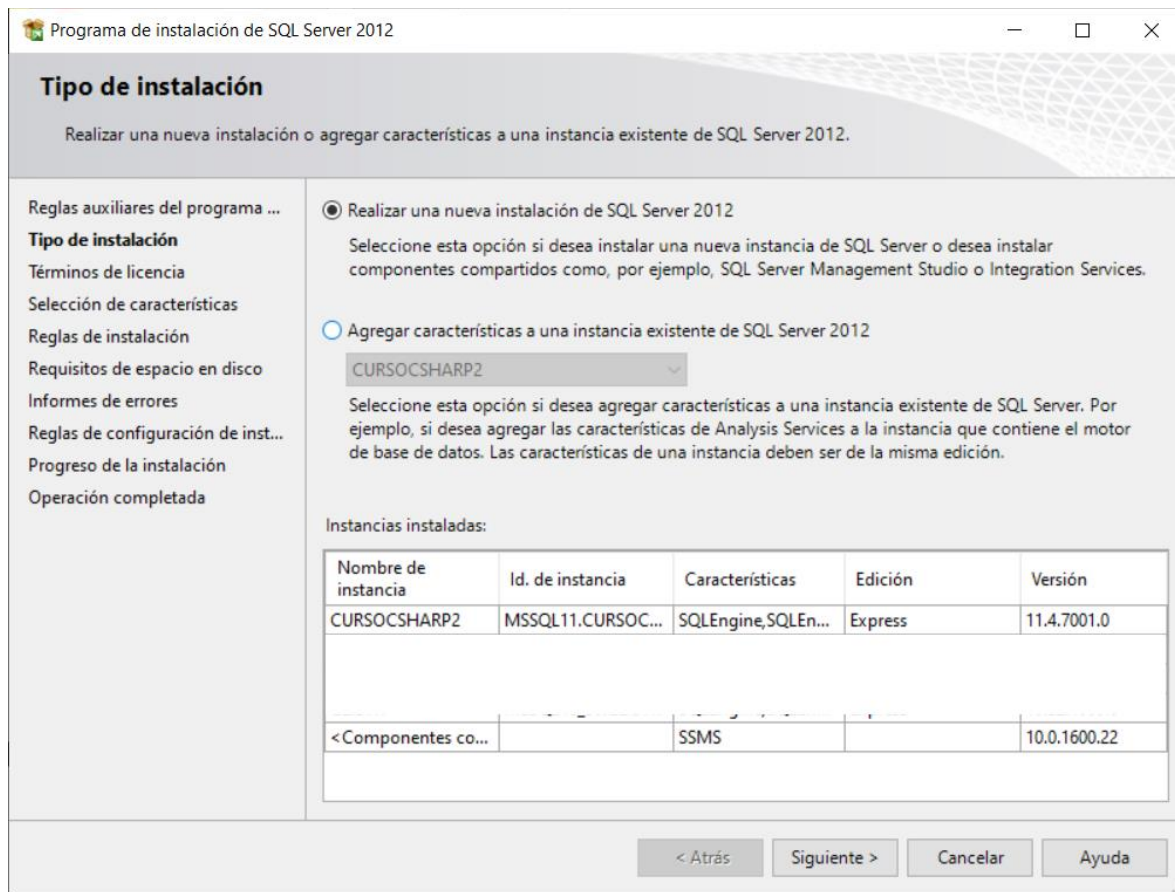


Fig. 73 – Instalación de SQL Server Management Studio

En este caso, no instalaremos todas las herramientas, sino que sólo instalaremos las herramientas de administración.

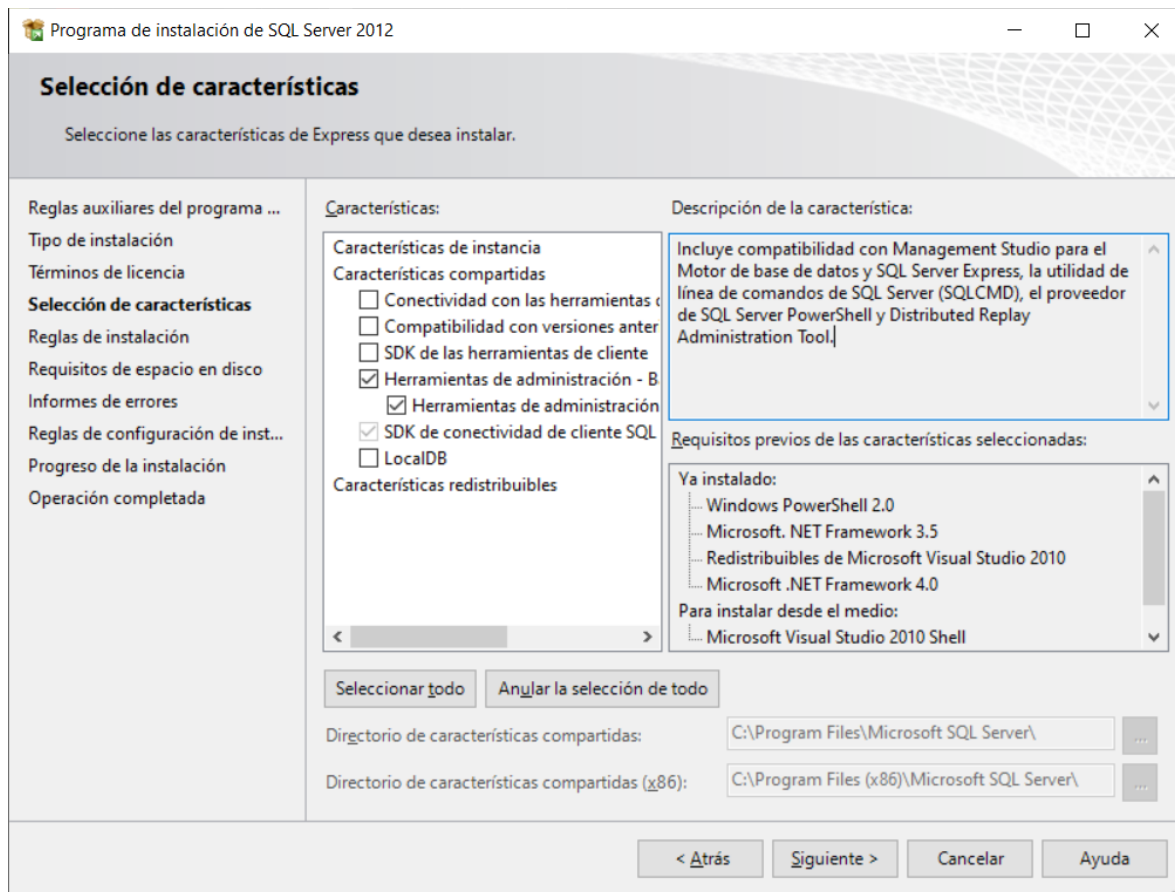


Fig. 74 – Características a instalar

Tras esto, la instalación procederá con normalidad, y al finalizar, podremos ejecutar el programa.



Fig. 75 – Pantalla de carga del SSMS 2012.

Al ejecutarse, tendremos la siguiente pantalla donde se nos pedirán las credenciales para conectarnos con el servidor. Ya que habilitamos ambos tipos de autenticación cuando realizamos la instalación, bien podemos seleccionar la autenticación de Windows o iniciar sesión con usuario y contraseña – el usuario será *sa* y la contraseña será la misma que hayamos definido anteriormente.

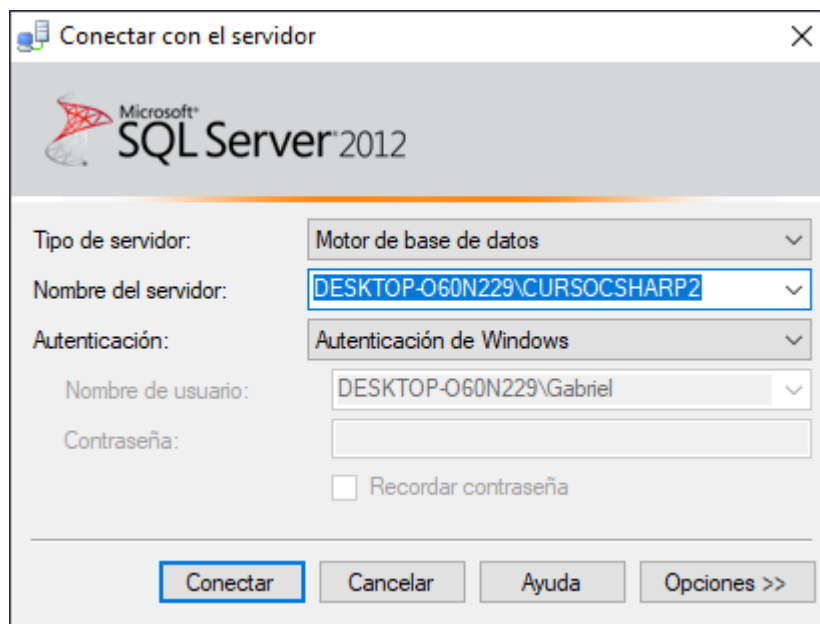


Fig. 76 – Autenticación en SSMS

Al acceder, tendremos vista a la pantalla de gestión de nuestro servidor en el cual podremos visualizar las bases de datos actualmente creadas, configurar nuestro servidor, crear nuevos usuarios, entre otros.

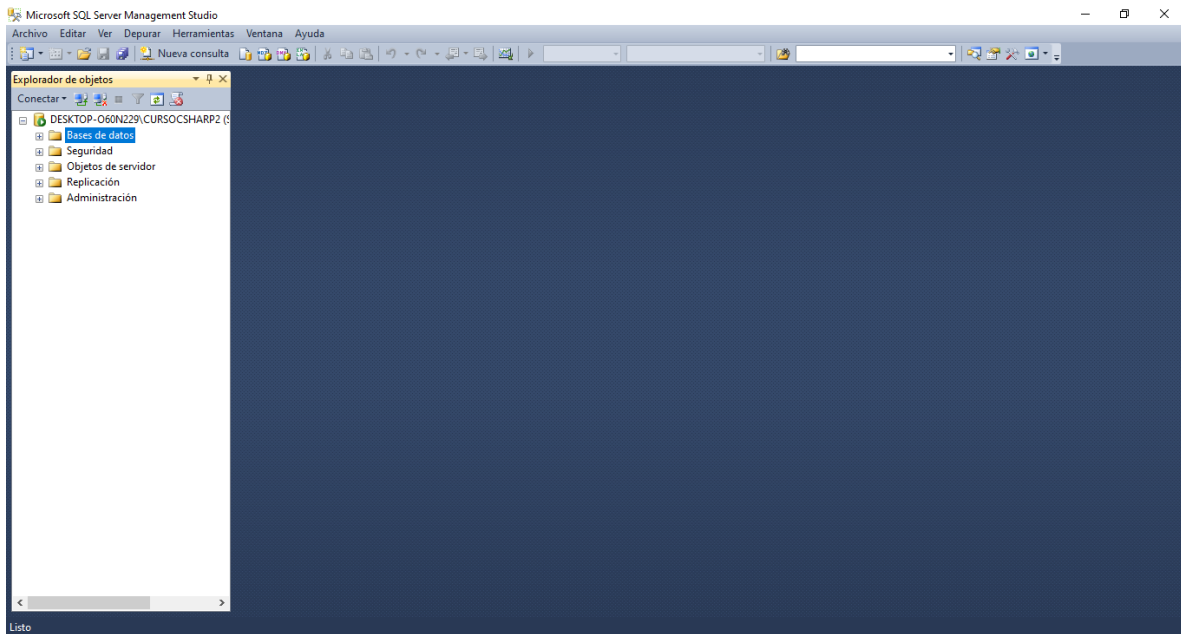


Fig. 77 – SQL Server Management Studio 2012 en ejecución.

Si quisiéramos, por ejemplo, crear un nuevo usuario, tendríamos que hacer doble click en Seguridad, y luego click derecho en la carpeta de Inicios de Sesión – Nuevo inicio de sesión.

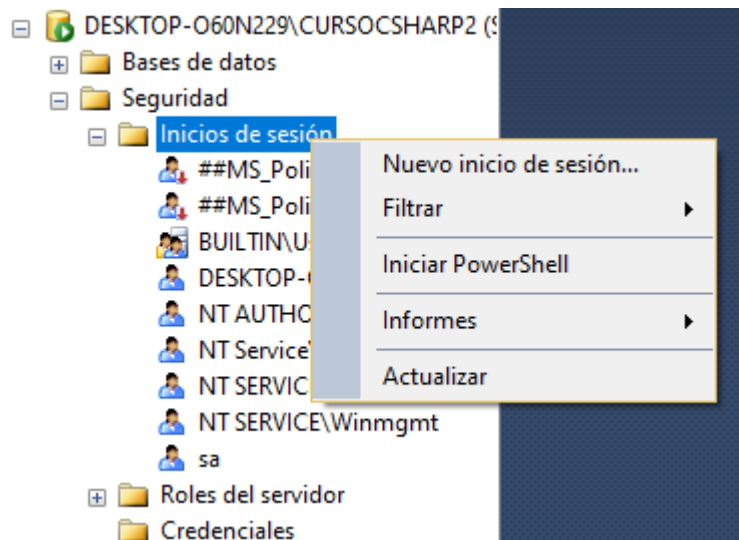


Fig. 78 – Creando un nuevo usuario

En esta nueva pantalla podremos introducir los datos de la nueva cuenta de usuario. En este caso, crearemos una cuenta llamada *curso* utilizando la autenticación de SQL Server.

Inicio de sesión - Nuevo

Script Ayuda

Selección de una página

- General
- Roles del servidor
- Asignación de usuarios
- Elementos protegibles
- Estado

Conexión

Servidor: DESKTOP-060N229\CURSOCSH

Conexión: sa

[Ver propiedades de conexión](#)

Progreso

Listo

Nombre de inicio de sesión: curso Buscar...

☐ Autenticación de Windows

☒ Autenticación de SQL Server

Contraseña: ..

Confirmar contraseña: ..

☐ Especificar contraseña anterior

Contraseña anterior: ..

☐ Exigir directivas de contraseña

☐ Exigir expiración de contraseña

☐ El usuario debe cambiar la contraseña en el siguiente inicio de sesión

☐ Asignado a certificado

☐ Asignado a clave asimétrica

☐ Asignar a credencial

Credenciales asignadas

Credencial	Proveedor
------------	-----------

Agregar


Quitar

Base de datos predeterminada: master

Idioma predeterminado: <predeterminado>

Aceptar Cancelar

Fig. 79 – Creación de un nuevo usuario

Tras esto, podremos desconectarnos haciendo click derecho al servidor – Desconectar, y volver a iniciar sesión con nuestra nueva cuenta haciendo click en Archivo -  Conectar Explorador de objetos...

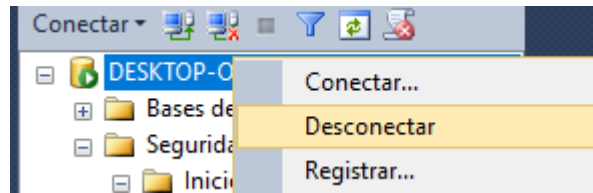


Fig. 80 – Desconectando el SSMS del servidor.

I.3 – Creación de Bases de Datos

Para crear una nueva base de datos, haremos click derecho sobre la carpeta Base de Datos – Nueva base de datos.

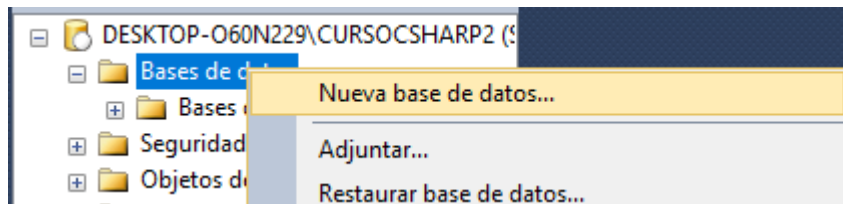


Fig. 80 – Creación de una nueva base de datos

Esto ejecutará una nueva ventana donde podremos indicar el nombre de nuestra nueva base de datos, además del nombre del usuario propietario de la misma. Crearemos una BD llamada *Ejemplo*, cuyo propietario será el usuario *curso* que recién creamos. Colocamos su nombre:

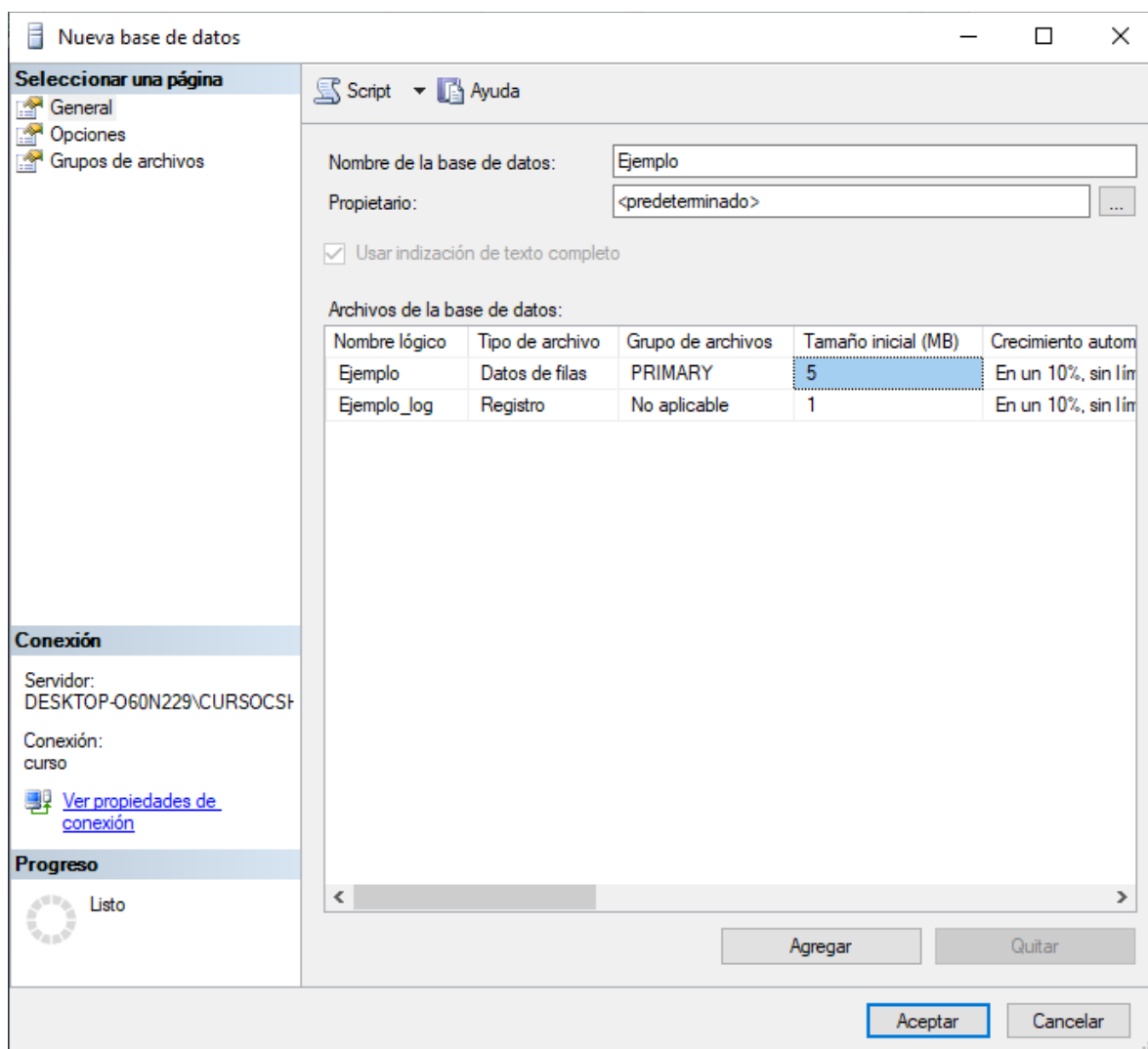


Fig. 81 – Creación de una nueva BD

Para seleccionar el propietario, haremos click al ícono de “...” situado a la derecha de esa caja de texto, y luego haremos click en Examinar.

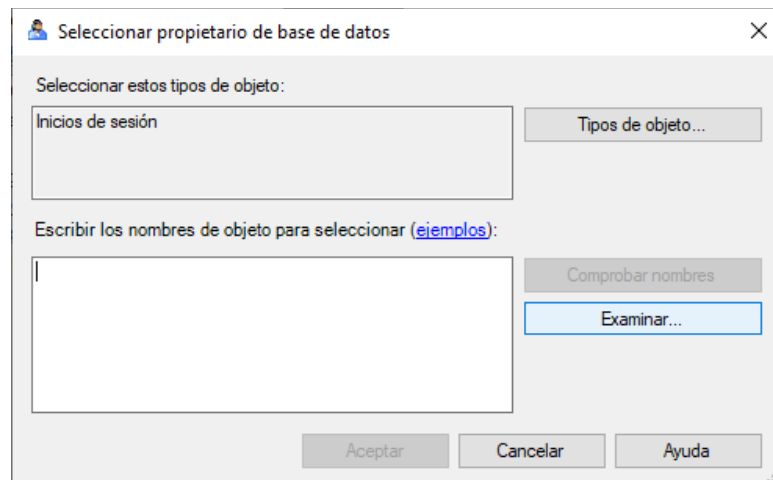


Fig. 82 – Seleccionando propietario de la BD

En la nueva pantalla, seleccionaremos la cuenta *curso* que acabamos de crear, haremos click en Aceptar – Aceptar, y tras esto, tendremos nuestra BD creada.

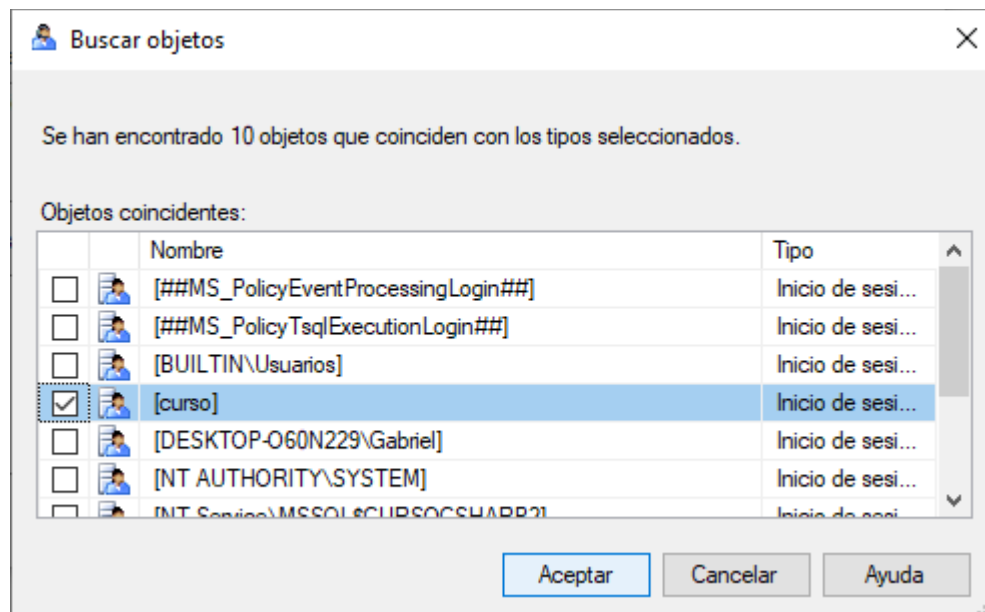


Fig. 83 – Selección de la cuenta curso como propietaria de la BD.

Es importante que la creación de la BD se ejecute en principio con la cuenta *sa*, puesto que al ser un superusuario, esta cuenta contará con la permisología necesaria para ejecutar la creación de la BD.

I.4 – Creación de Tablas

Para crear una tabla, debemos abrir nuestra BD, y luego hacer click derecho en la carpeta Tablas – Nueva Tabla.

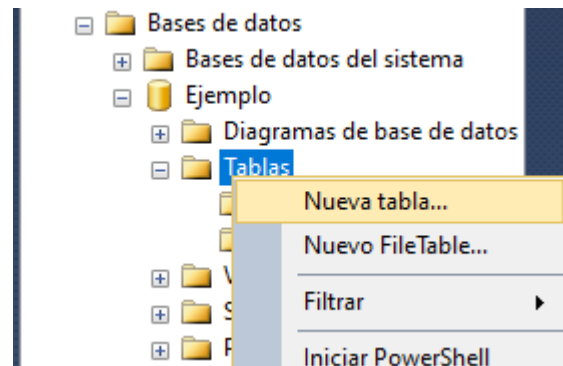


Fig. 84 – Selección de opción para crear nueva tabla

La ventana cambiará a una en que se nos permitirá configurar la tabla, agregar nuevas columnas y modificar las propiedades de las mismas. Para modificar el nombre de la tabla, debemos ir a la pestaña de Propiedades, en la cual podremos modificar su Nombre. Para agregar nuevas columnas, sólo es necesario introducir el nombre de la misma, y luego seleccionar el tipo de dato a la derecha. Las propiedades se pueden configurar en la pestaña inferior.

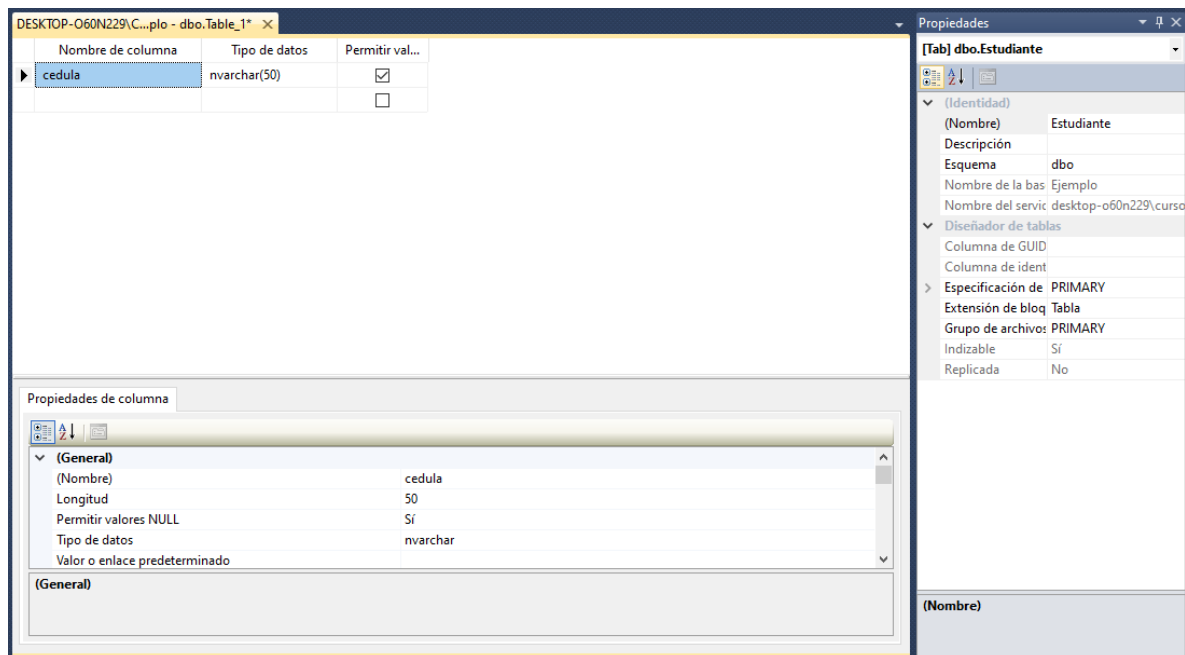



Fig. 85 – Creación de una tabla

Una vez que esté configurada la totalidad de las columnas de la tabla, sólo basta con hacer click en el botón  para finalizar su creación. Tras esto, podremos hacer click derecho en Tablas – Actualizar, lo que desplegará nuestra nueva tabla recién creada.

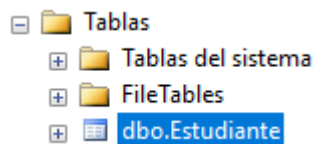


Fig. 86 – Tabla recién creada

I.4.1 – Relaciones y tipos de datos

Ejercicio Propuesto

Spider-Men Creative Studio desea desarrollar una aplicación que les permita gestionar la información de toda su Spider-People. Para ello, desean que la aplicación implemente una base de datos en SQL Server. Considerando que los distintos Spider-Men pertenecen a distintos universos, y que no hay limitante alguna para ellos – un universo puede contar con diferentes Spider-Man, ellos decidieron realizar un análisis de villanos, con el cual concluyeron que los villanos en esencia siempre son los mismos, pero cuentan con diferencias sutiles entre universos – Por ejemplo, el Dr. Otto Octavius es la Dra. Olivia Octavius en la Tierra-18, a pesar de compartir sus alter egos de Doctor Octopus.

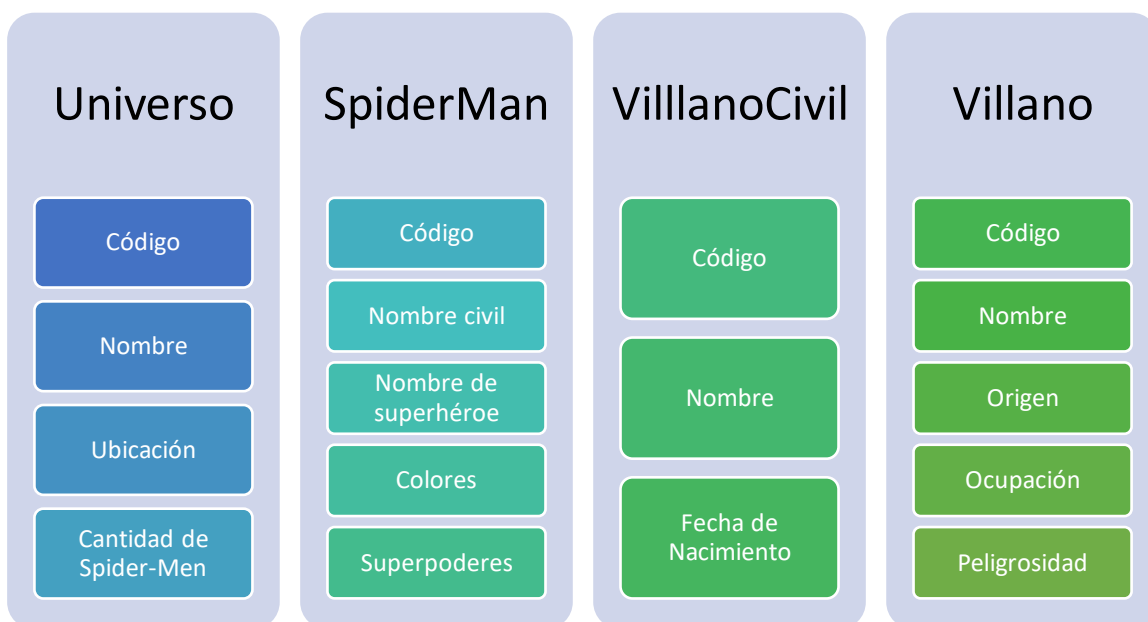
Así, el programa debe gestionar las siguientes entidades:

- **Universos:** En la cual se guardará su código identificador único, su nombre, la ubicación del universo en el espacio-tiempo y la cantidad máxima de Spider-Men que pueden existir en él
- **Spider-Men:** Se guardará su nombre civil, su nombre de superhéroe, sus colores y sus superpoderes, además de estar vinculado a un universo en específico.
- **Villanos:** De cada uno se debe guardar su información civil y su información de villano. Entre su información civil está su código identificador único, su nombre civil, su fecha de nacimiento y su universo.

Entre su información de villano está su código identificador único, su nombre de villano, su origen, su ocupación y su nivel de peligrosidad, del 1 al 10.

A destacar que la información civil de un villano debe estar vinculada a únicamente una información de villano, y que todas las entidades deben tener un estatus. Realiza el diseño de la base de datos para solucionar este problema, e implementala usando SQL Server.

Necesitamos establecer primero nuestras entidades y sus relaciones. En principio, debemos saber que el programa almacena información de Universos, Spider-Men, Villanos en su identidad civil y Villanos en su identidad de villano. Cada una de estas entidades tendrá su propia información, reflejados en el cuadro a continuación:

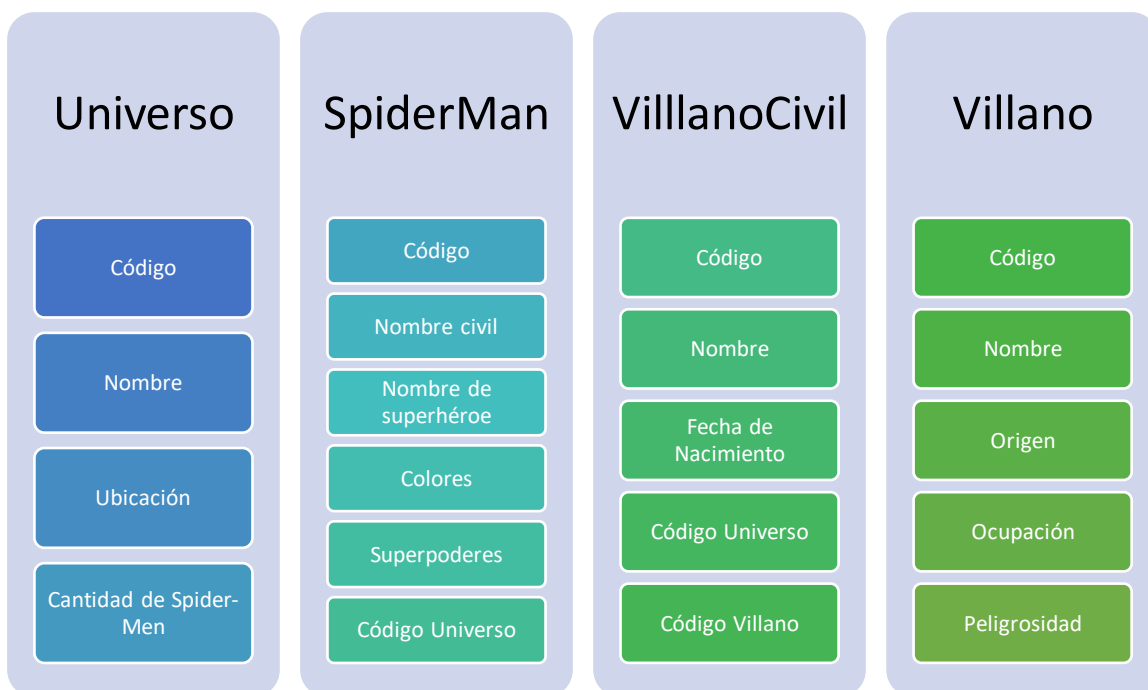


Cuadro 1 – Entidades del ejercicio, sin relaciones

Para establecer las relaciones, debemos entender que:

- 1) Un Spider-Man pertenece a un único universo, y un Universo puede tener muchos Spider-Man.
- 2) Un VillanoCivil pertenece a un único universo, y un Universo puede tener muchos VillanosCivil.
- 3) Un VillanoCivil pertenece a un único Villano, y un Villano puede tener muchos VillanosCivil.

Así, podemos visualizar las siguientes entidades con sus relaciones:



Cuadro 2 – Entidades del ejercicio, con relaciones

Todo esto queda reflejado en el siguiente diagrama MER:

MER - Spider-Men Creative Studio

GracoSoft - C# II | Enero 2022

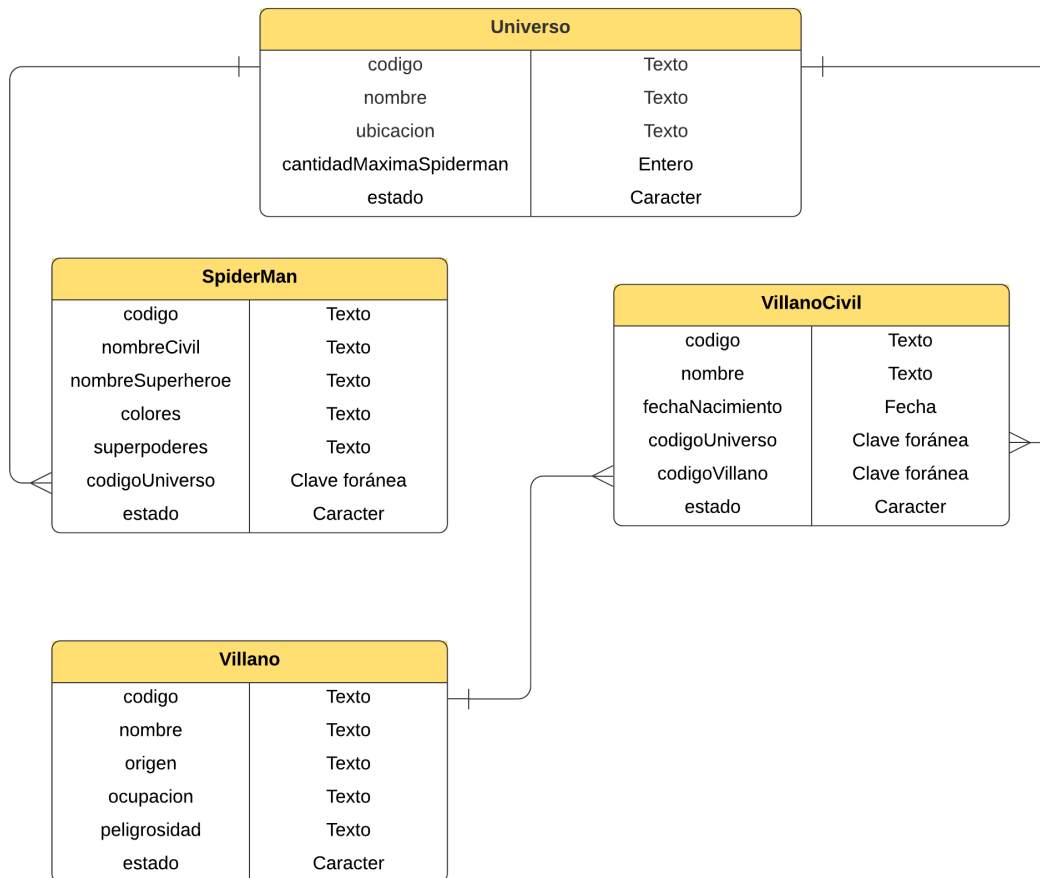


Diagrama 1 – MER para Spider-Men Creative Studio, con tipos de datos genéricos

Ahora bien, entendiendo que cada uno de estos campos deben tener su tipo de dato en específico, vamos a estudiar los tipos de datos más comunes en SQL Server.

Tipo de dato	Valor Mínimo	Valor Máximo	Equivalente en C#	Nota
Int	-2.147.483.648	2.147.486.648	int	Números enteros sin parte decimal
Decimal	$-10^{38}+1$	$10^{38}-1$	double, float, decimal	Números enteros con parte decimal
Datetime	1753-01-01 00:00:00.000	9999-12-31 23:59:59.997	Datetime	Fecha y hora en conjunto
Date	0001-01-01	9999-12-31	Date	Sólo fecha
Time	00:00:00.0000000	23:59:59.9999999	Time	Sólo hora
Datetime2	0001-01-01 00:00:00.0000000	9999-12-31 23:59:59.9999999	Datetime	Combina el tipo de dato Date y Time. Mayor exactitud en las horas.
Char	0 caracteres	8000 caracteres	char	Longitud fija, todos los valores deben tener la misma longitud.
Varchar	0 caracteres	8000 caracteres	char, string	Texto de longitud variable, no define longitud máxima.

Varchar(max)	0 caracteres	2^31 caracteres	char, string	Texto de longitud variable, longitud máxima explícitamente definida.
Varbinary(max)	0 bytes	2^31 bytes	bytes	Cualquier archivo, imagen, elemento en binario.

Tabla 1 – Tipos de datos más comunes en SQL Server 2012

Sabiendo esto, podemos redefinir el MER, esta vez estableciendo los tipos de datos según SQL Server. A destacar que, en este caso, se va a indicar también cuáles son los campos que componen las claves, tanto las claves primarias como las claves foráneas, utilizando *pk* para las claves primarias (del inglés *primary key*), y *fk* para las claves foráneas (del inglés *foreign key*).

El resultado es el siguiente diagrama MER:

MER - Spider-Men Creative Studio

GracoSoft - C# II | Enero 2022

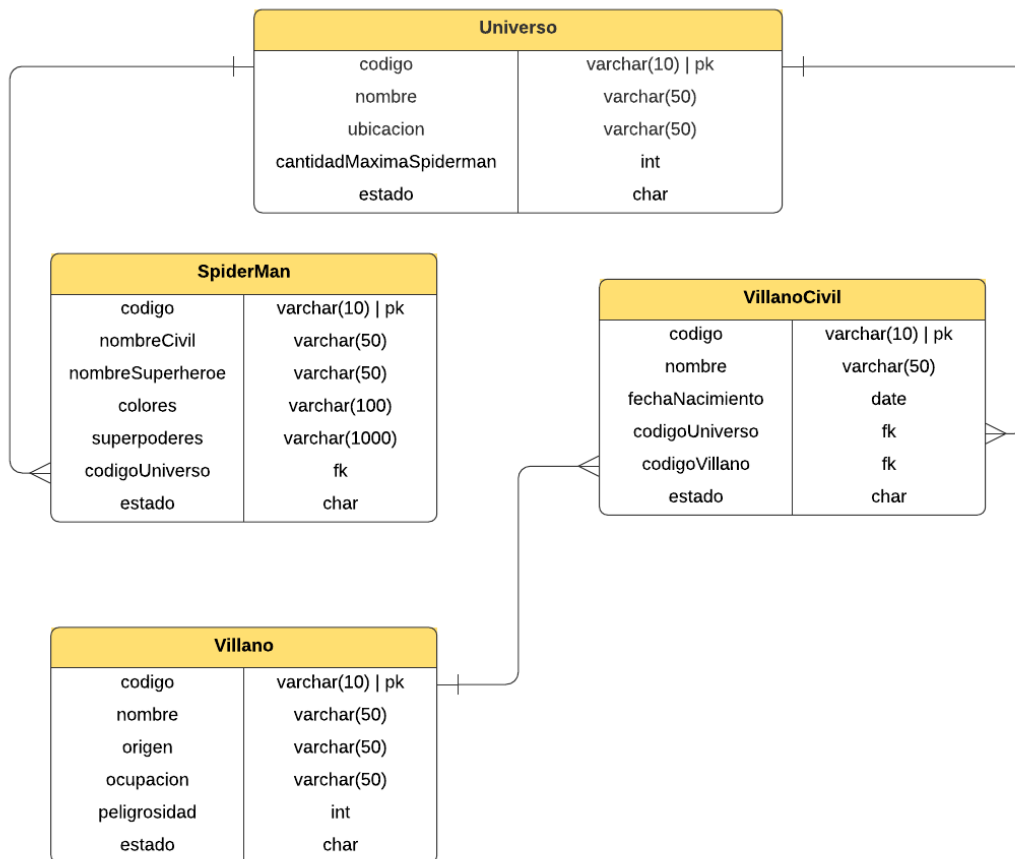


Diagrama 2 – MER para Spider-Men Creative Studio, con tipos de datos para SQL Server 2012

Con esto podemos proceder a crear cada una de las tablas. Comenzaremos con la tabla Universo, puesto que esta es la tabla central a partir de la cual derivan las demás.

Crearemos la tabla, que nombraremos Universo, y agregaremos el campo código, tipo de dato varchar(10).

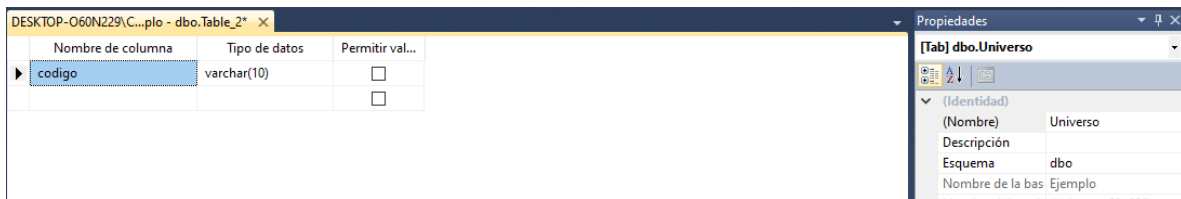


Fig. 87 – Campo código en la tabla Universo

Para definirlo como la clave primaria, haremos click derecho sobre este campo que acabamos de crear – Establecer clave principal:

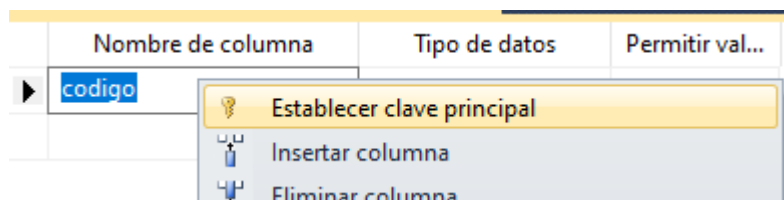


Fig. 88 – Asignando clave principal de la tabla Universo

Y, una vez teniendo la clave primaria, podemos proceder con los demás campos.

Nombre de columna	Tipo de datos	Permitir val...
codigo	varchar(10)	<input type="checkbox"/>
nombre	varchar(50)	<input checked="" type="checkbox"/>
ubicacion	varchar(50)	<input checked="" type="checkbox"/>
cantidadMaximaSpider...	int	<input checked="" type="checkbox"/>
estado	char(1)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Fig. 89 – Campos creados de la tabla Universo

Procedemos con la tabla SpiderMan de manera análoga, definiendo su clave principal. En este caso, el campo codigoUniverso se trabajará igualmente como un varchar(10).

	Nombre de columna	Tipo de datos	Permitir val...
🔑	codigo	varchar(10)	<input type="checkbox"/>
	nombreCivil	varchar(50)	<input checked="" type="checkbox"/>
	nombreSuperheroe	varchar(50)	<input checked="" type="checkbox"/>
	colores	varchar(100)	<input checked="" type="checkbox"/>
	superpoderes	varchar(1000)	<input checked="" type="checkbox"/>
	codigoUniverso	varchar(10)	<input type="checkbox"/>
	estado	char(1)	<input checked="" type="checkbox"/>

Fig. 90 – Tabla SpiderMan

Para definir la clave foránea, haremos click derecho – Relaciones, y a continuación, haremos click en el botón Agregar.

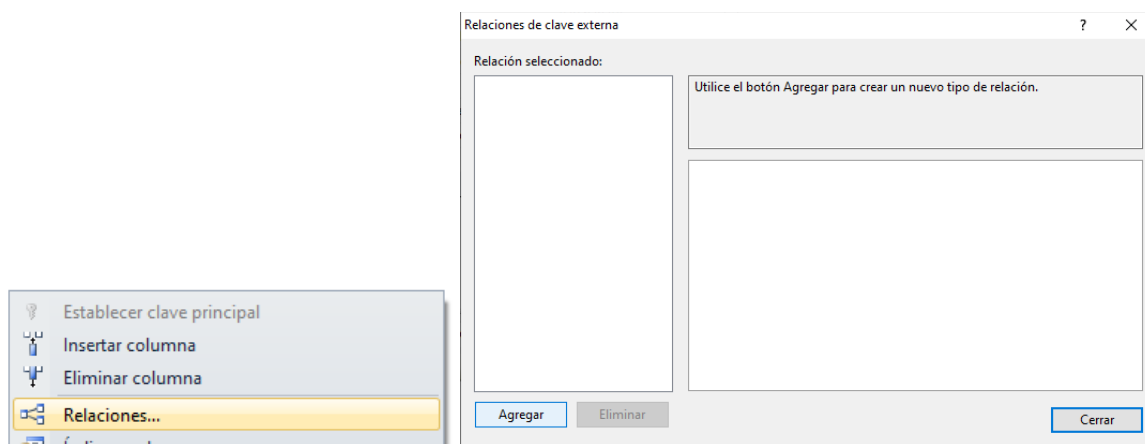


Fig. 91 – Agregando clave foránea

Luego, haremos click en la pestaña de Especificación de tablas y columnas, para hacer click en el botón de “...” situado a la derecha.

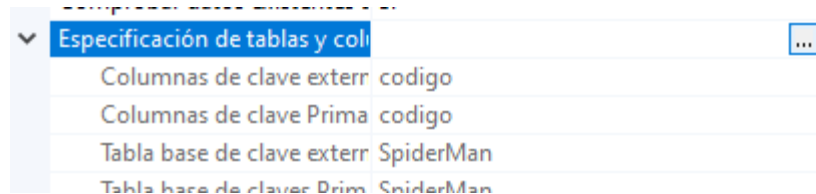


Fig. 92 – Pantalla previa a la selección de las tablas relacionadas

En esta nueva ventana, seleccionaremos la tabla Universo, y luego haremos click en el recuadro blanco para seleccionar el campo código. Del mismo modo, en el recuadro a su derecha, seleccionaremos el campo codigoUniverso, lo que resultará con la siguiente ventana:

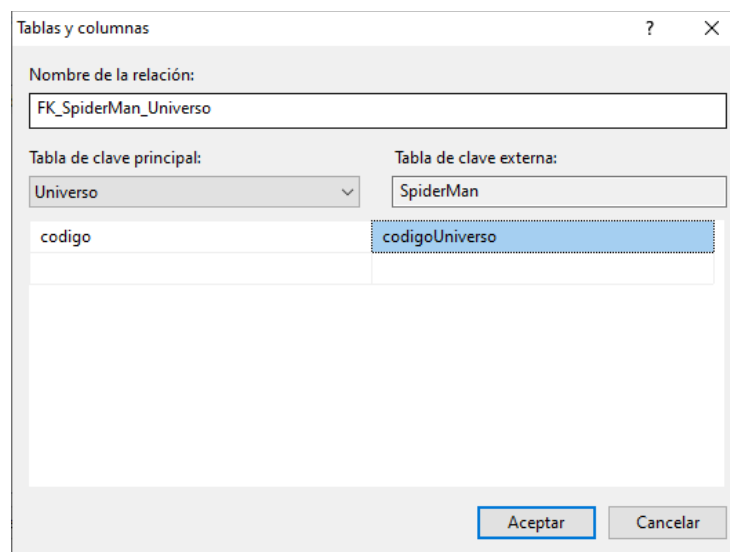


Fig. 93 – Configurando clave foránea

Luego, hacemos click en Aceptar – Cerrar – Guardar y listo, tendremos nuestra segunda tabla.

Para las demás tablas se opera de manera análoga y su creación se deja asignada al lector.

I.4.2 – Modificación de tablas

Asumamos ahora que queremos modificar la tabla SpiderMan para anexar un nuevo campo llamado Ciudad, en el cual indicaremos cuál es la ciudad en la que opera un SpiderMan en particular. Para modificar la tabla, haremos click derecho sobre la tabla – Diseño.

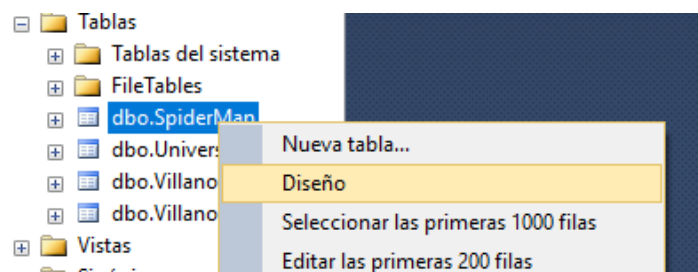


Fig. 94 – Selección de opción para modificar una tabla

Tras esto, volveremos a la pantalla anteriormente vista en que podremos modificar nuestra tabla. Bien podemos añadir la nueva columna introduciendo el texto en la última celda, o bien podemos hacer click derecho – Insertar columna, acción que anexará la columna inmediatamente después de la columna que tengamos seleccionada.

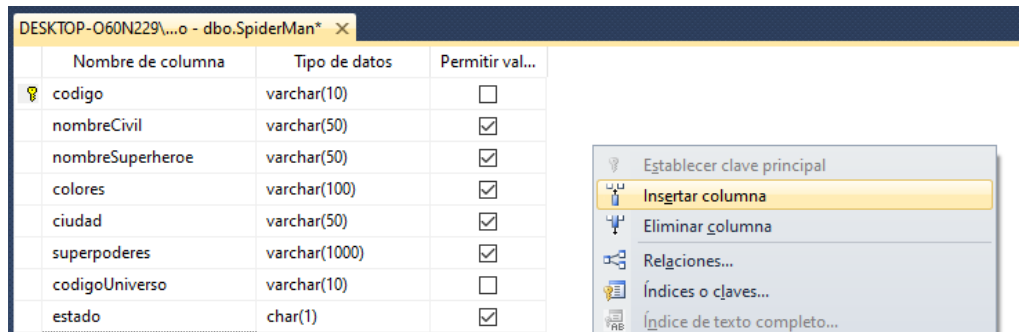



Fig. 95 – Insertar una columna

Podemos también modificar el nombre de las columnas haciendo doble click sobre la celda, y podemos eliminar una columna al hacer click derecho en la misma – Eliminar columna. En todo caso, basta con hacer click en el botón  para guardar los cambios realizados.

I.4.3 – Generación de Diagrama MER

Si tenemos una base de datos ya diseñada, también podemos aprovechar SSMS para generar un diagrama MER asociado a la misma. Para ello, hacemos click derecho en la carpeta Diagramas de bases de datos – Nuevo diagrama de bases de datos:

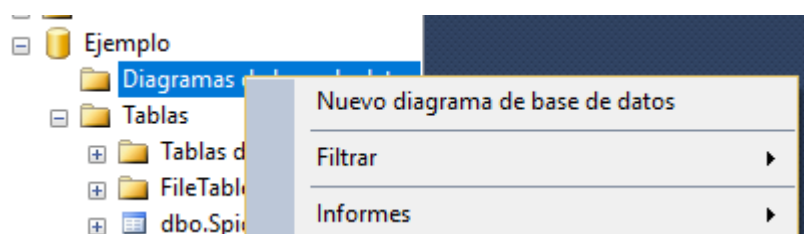


Fig. 96 – Generando un diagrama MER.

Al hacerlo, nos aparecerá una ventana con todas las tablas. Hacemos click en agregar para cada una de las tablas y luego en Cerrar, para finalmente tener la vista del diagrama.

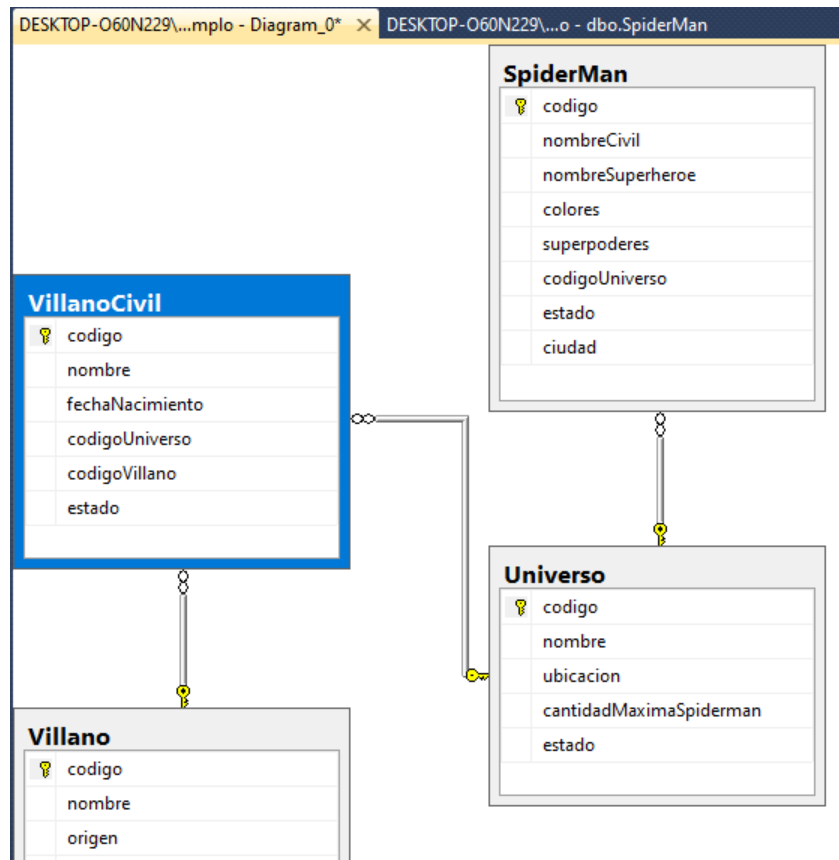


Fig. 97 – Diagrama MER visto en SSMS



Si bien SSMS no permite exportar nativamente un diagrama MER, sí permite copiarlo al portapapeles, para lo cual debemos hacer click derecho sobre el diagrama – Copiar diagrama al portapapeles. Tras esto, podremos ir a cualquier software de edición gráfica o de ofimática y pegar el diagrama exportado.



Fig. 98 – Diagrama MER exportado

I.5 – Ejecución de sentencias

Evidentemente SQL Server, al ser un manejador de bases de datos basado en SQL, permite la ejecución de sentencias para añadido, modificación, agregación, ordenamiento, agrupamiento, entre muchas otras acciones. Estas sentencias son, en esencia, muy parecidas a las sentencias escritas para otros manejadores como PostgreSQL, MySQL u Oracle, con algunas diferencias sutiles que no forman parte del alcance de este curso.

Para ejecutar una sentencia desde el SSMS, sólo hace falta hacer click en el botón  Nueva consulta, lo que abrirá una ventana con un editor de texto que podremos utilizar para escribir nuestras sentencias. Para ejecutarlas, podemos hacer click en el botón  Ejecutar o en su defecto presionando la tecla F5. Ahora bien, asumamos la siguiente sentencia SQL Insert:

A screenshot of a SQL query editor window with a dark background and a purple border. The window contains an SQL INSERT statement. The text is as follows:

```
INSERT INTO [dbo].[Universo]
    ([codigo]
    ,[nombre]
    ,[ubicacion]
    ,[cantidadMaximaSpiderman]
    ,[estado])
VALUES ('16',
    'Tierra-16',
    'Vía láctea, 2020 D.C.',
    1,
    'A')
```

Fig. 99 – Sentencia SQL para insertar un Universo

Podemos notar cómo SQL Server utiliza los corchetes [] para indicar nombres propios de objetos: bien sea un esquema [dbo], una tabla [Universo], o una columna [código], SQL Server lo encierra en corchetes en lugar de utilizar comillas dobles, mientras que los valores siguen encerrados en comillas simples “.

Si ejecutamos la sentencia, tendremos el siguiente mensaje en la región inferior de la pantalla:

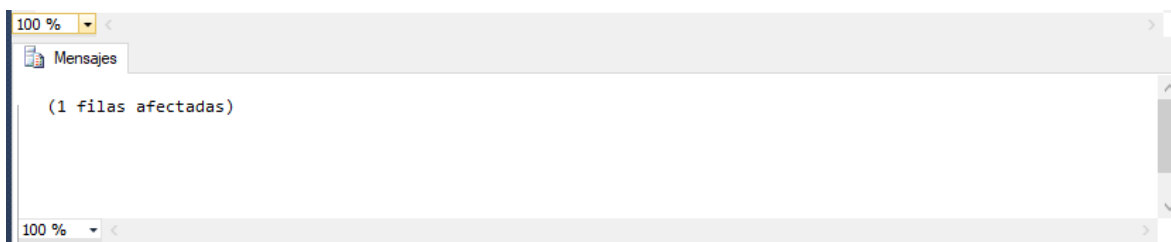


Fig. 100 – Mensaje de éxito tras ejecutar una sentencia SQL

Luego, si ejecutamos por ejemplo el siguiente SELECT:



Fig. 101 – SELECT para verificar el agregado del universo

En esa misma zona de la pantalla se desplegarán los resultados de la sentencia.

	codigo	nombre	ubicacion	cantidadMaximaSpideman	estado
1	15	Tierra-15	Andromeda, galaxia espiral, 9453 A.C.	3	A
2	16	Tierra-16	Via láctea, 2020 D.C.	1	A

Fig. 102 – Resultados de un SELECT

SSMS también nos ofrece las siguientes utilidades al hacer click derecho en una tabla:

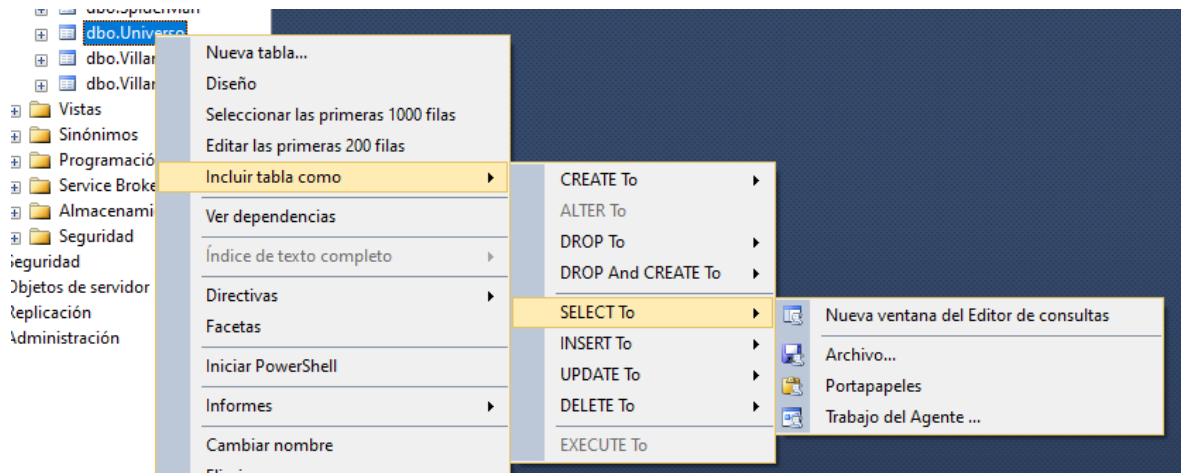
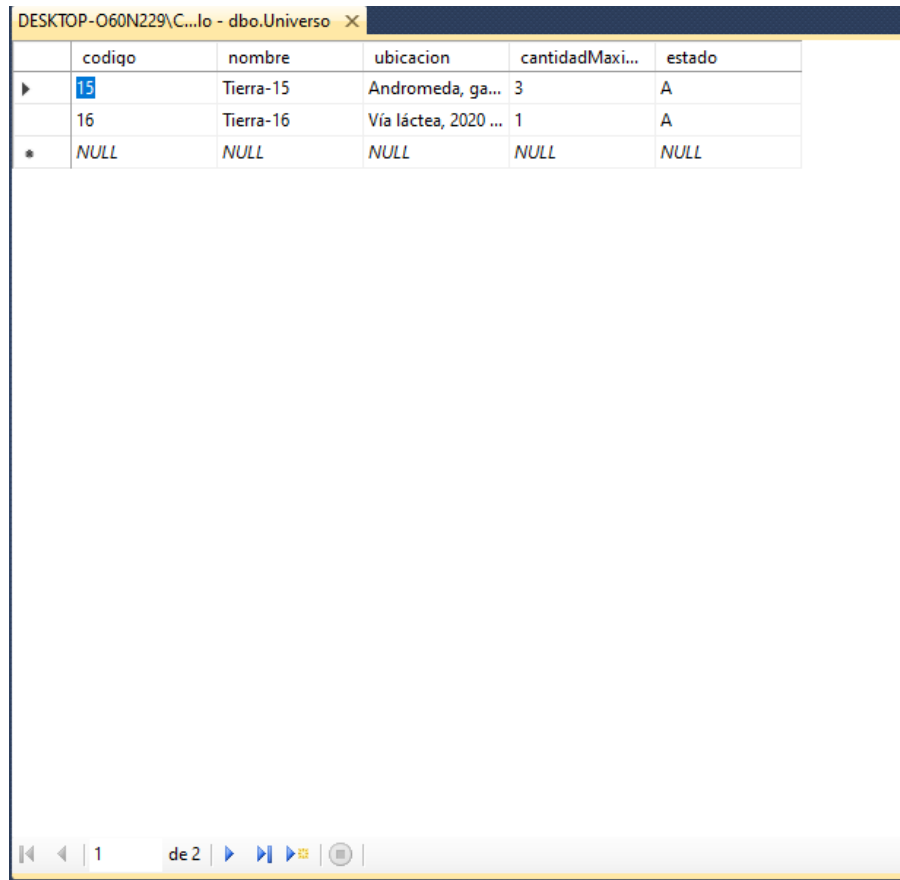


Fig. 103 – Utilidades de SSMS con una tabla

- Nos permite realizar un SELECT con las primeras 1000 filas sin necesidad de escribir la sentencia SQL.
- Nos permite generar automáticamente las sentencias para realizar SELECT, INSERT, UPDATE y DELETE y copiarlas al portapapeles, guardarlas a un archivo o abrir una ventana de sentencia con la misma.

Y, por último, nos brinda la opción de editar las primeras 200 filas. Al hacer click en ella, nos aparecerá una ventana con los datos editables de las primeras 200 filas de una tabla en particular.



	codigo	nombre	ubicacion	cantidadMaxi...	estado
▶	15	Tierra-15	Andromeda, ga...	3	A
	16	Tierra-16	Vía láctea, 2020 ...	1	A
*	NULL	NULL	NULL	NULL	NULL

Fig. 104 – Editando datos de una tabla

Debemos ser muy cuidadosos a la hora de realizar cualquier tipo de modificación o alteración sobre estas tablas, dado que todos los cambios pasan en vivo a la base de datos sin ningún tipo de control transaccional con tan sólo presionar **Enter**, lo que implica que los cambios realizados sobre estos datos **no se pueden deshacer**.

I.6 – Ejercicios propuestos

1) Implemente una base de datos, llamada Hospital que contenga las siguientes tablas y columnas:

a. Médico

- i. Código – Texto
- ii. Número de Cédula de Identidad – Texto
- iii. Nombres y apellidos completos – Texto
- iv. Fecha de nacimiento – Fecha
- v. Especialidad – Clave foránea con Especialidad.
- vi. Estado – Caracter.

b. Especialidad.

- i. Código – Texto
- ii. Nombre – Texto
- iii. Parte(s) del cuerpo que estudia – Texto
- iv. Cantidad necesaria en el hospital – Número
- v. Estado – Caracter

c. Paciente.

- i. Cédula de Identidad – Texto
- ii. Nombres y apellidos completos – Texto
- iii. Fecha de nacimiento – Fecha
- iv. Última visita – Fecha
- v. Estado – Caracter

d. Dolencia

- i. Código – Texto
- ii. Nombre – Texto

- iii. Especialidad que la atiende – Clave foránea con Especialidad
 - iv. Gravedad – Número
 - e. Consulta
 - i. Código – Texto
 - ii. Cédula del paciente – Clave foránea con Paciente
 - iii. Código del doctor – Clave foránea con Médico
 - iv. Fecha – Fecha
 - f. Dolencias por Consulta
 - i. Código de la consulta – Clave foránea con Consulta
 - ii. Código de la dolencia – Clave foránea con Dolencia
- 2) Llena la base de datos con datos anterior con datos de prueba. Tras esto, escriba las sentencias SQL que extraigan la siguiente información:
 - a. Dolencias por un paciente dado.
 - b. Consultas por un médico.
 - c. Consultas por especialidad.
 - d. Dolencias diagnosticadas por un médico.
 - e. Listado de médicos con más de 10 consultas en un día dado.
- 3) La E3V, o Exposición de Equipos Electrónicos Venezolanos, es una conferencia nacional que se lleva a cabo cada año. En esta, los fabricantes venezolanos muestran todos los equipos que van a lanzar en el año siguiente, además de promocionar sus últimos avances tecnológicos y científicos. La exposición cuenta con las siguientes características:

- a. Cada empresa debe registrarse con sus datos particulares: RIF, razón social, área a la que se dedica. Además de eso, en los registros de la E3V también se tiene cuándo fue su última participación.
- b. Cada empresa debe registrar con antelación los productos que van a mostrar en la exposición. En este registro, se genera un código único automáticamente, además de tomar el nombre del producto, la categoría a la que pertenece, la fecha estimada de comercialización y una foto del mismo.
- c. Como es una exposición anual, cada año se debe crear un registro de la misma. En este registro se guarda un código, el número del año, la fecha exacta en que se ejecutó y la ciudad que la hospedó.
 - i. En este registro, también se deben guardar todas las empresas que participaron, junto a los productos que mostraron.

Diseñe e implemente una Base de Datos en SQL Server que permita registrar todos estos datos.

II – Referencias Bibliográficas

ConnectionStrings. (07 de 01 de 2022). *SQL Server 2012 Data Types Reference* - *ConnectionStrings.com*. Obtenido de ConnectionStrings: <https://www.connectionstrings.com/sql-server-2012-data-types-reference/>

Farrell, J. (2018). *Microsoft Visual C# 2017: An Introduction to Object-Oriented Programming*. Boston: Cengage Learning.

Griffiths, I. (2019). *Programming C# 8.0*. Sebastopol: O'Reilly Media, Inc.

Microsoft. (28 de 01 de 2021). *C# documentation*. Obtenido de Microsoft Docs: <https://docs.microsoft.com/en-us/dotnet/csharp/>

Pérez, S. D. (07 de 01 de 2022). *¿Qué es Microsoft SQL Server y para qué sirve?* Obtenido de Intelequia: <https://intelequia.com/blog/post/2948/qu%C3%A9-es-microsoft-sql-server-y-para-qu%C3%A9-sirve>

Siahaan, V. (2020). *VISUAL C# .NET FOR STUDENTS: A Project-Based Approach to Develop Desktop Applications*. Balige: BALIGE PUBLISHING.

Siahaan, V., & Sianipar, R. (2020). *VISUAL C# .NET: A Step By Step, Project-Based Guide to Develop Desktop Applications*. Balige: BALIGE PUBLISHING.

Villalobos, J., & Casallas, R. (s.f.). *Fundamentos de Programación. Aprendizaje activo basado en casos*. Bogotá: Universidad de los Andes - Facultad de Ingeniería.

