



INSTITUTO TECNOLÓGICO  
DE CIUDAD MADERO

**ITCM**



**SEP**

**Materia: Administración de Base de Datos**

**Proyecto Final**

**“Ciclo de vida de una base de datos”**

**Alumno: Francisco Gabriel Reyna  
Montaño**

**No. de Control: 22070272**

**Fecha: 1/6/2025**

# Introducción

Este informe se creó para detallar las actividades hechas por la asignación de trabajo dada como proyecto final de la materia. El objetivo fue aplicar todos los conocimientos vistos en la materia sobre la administración de SQL Server, abarcando desde la creación y diseño de una base de datos para un sistema de expediente clínico basado en la norma; NOM-004-SSA3-2012, con énfasis en los puntos 5 y 6. En este informe se ve sobre la configuración de seguridad, la creación de tareas de respaldo automatizadas, la implementación de planes de mantenimiento, la modificación de la base de datos y, finalmente, su restauración a un estado previo. Todas las tareas se realizaron utilizando SQL Server Management Studio (SSMS).

## 1.- Creación de la Base de Datos

Se procedió a crear el esquema de la base de datos denominada *HospitalDB*, diseñada para el registro de expedientes clínicos digitales. El diseño se basó en los requerimientos de la Norma Oficial Mexicana NOM-004-SSA3-2012, específicamente en los puntos 5 que habla de “generalidades”, y 6 que es: “Del expediente clínico en consulta general y de especialidad”.

### Script SQL para la Creación de la Base de Datos (HospitalDB):

```
CREATE DATABASE [HospitalDB];
GO
USE [HospitalDB];
GO
-- Tabla para Establecimientos (5.2.1, 5.2.2)
CREATE TABLE Establecimientos (
    DomicilioEstablecimiento NVARCHAR(MAX) NOT NULL,
    NombreInstitucionPertenece NVARCHAR(255), -- Ejemplo: ISSTE, IMSS, Privado
    RazonSocialPropietario NVARCHAR(255), -- (Ref: 5.2.2)
    DenominacionSocialPropietario NVARCHAR(255)
);
GO
-- Tabla para Pacientes (5.2.3)
CREATE TABLE Pacientes (
    EstablecimientoID INT PRIMARY KEY IDENTITY(1,1),
    TipoEstablecimiento NVARCHAR(100),
    NombreEstablecimiento NVARCHAR(255) NOT NULL,
```

```

    PacienteID INT PRIMARY KEY
    IDENTITY(1,1),

    NombreCompleto NVARCHAR(255) NOT
    NULL,

    Sexo NVARCHAR(10) NOT NULL, -- Ejemplo:
    Femenino, Masculino, otros

    FechaNacimiento DATE,

    Edad AS ( -- Edad calculada en automatico

        CASE

            WHEN FechaNacimiento IS NULL THEN
            NULL

            WHEN DATEADD(YEAR,
            DATEDIFF(YEAR, FechaNacimiento,
            GETDATE()), FechaNacimiento) > GETDATE()

            THEN DATEDIFF(YEAR,
            FechaNacimiento, GETDATE()) - 1

            ELSE DATEDIFF(YEAR,
            FechaNacimiento, GETDATE())

            END

        ),

    DomicilioPaciente NVARCHAR(MAX) NOT
    NULL,

    GrupoEtnico NVARCHAR(100),

    FechaRegistro DATETIME2 DEFAULT
    GETDATE()

);

GO

```

```

CREATE TABLE PersonalSalud (

    PersonalID INT PRIMARY KEY
    IDENTITY(1,1),

```

```

    NombreCompleto NVARCHAR(255) NOT
    NULL,

```

```

    CedulaProfesional VARCHAR(50), -- Si es
    que aplica

```

```

    Especialidad NVARCHAR(100), -- Ejemplos:
    Cardiologia, Pediatria

```

```

    TipoPersonal NVARCHAR(50) NOT NULL --
    Ejemplos: Medico, Enfermera, Técnico
    Laboratorista

```

```

);

GO

```

```

CREATE TABLE HistoriasClinicas (

```

```

    HistoriaClinicaID INT PRIMARY KEY
    IDENTITY(1,1),

```

```

    PacienteID INT NOT NULL,

```

```

    EstablecimientoID INT NOT NULL,

```

```

    PersonalSaludID INT NOT NULL,

```

```

    FechaHoraCreacion DATETIME2 NOT NULL,

```

```

    FirmaCreador NVARCHAR(255),

```

```

    EdadPacienteAlMomento INT,

```

```

    SexoPacienteAlMomento NVARCHAR(10), --
    Sexo del paciente al momento (5.9)

```

```

    NumeroCamaExpediente NVARCHAR(50), --
    Numero de cama o expediente (5.9), no se
    requiere si es consulta general/especialidad

```

```

-- 6.1.1 Interrogatorio

```

```

    GrupoEtnico NVARCHAR(100),

```

AntecedentesHeredoFamiliares  
NVARCHAR(MAX),

AntecedentesPersonalesPatologicos  
NVARCHAR(MAX), -- Incluir uso y dependencia  
de tabaco, alcohol, drogas, otros

AntecedentesPersonalesNoPatologicos  
NVARCHAR(MAX),

PadecimientoActual NVARCHAR(MAX),

InterrogatorioAparatosSistemas  
NVARCHAR(MAX),

-- 6.1.2 Exploración Fisica

HabitusExterior NVARCHAR(MAX),

Temperatura DECIMAL(4,1),

TensionArterial VARCHAR(20),

FrecuenciaCardiaca INT,

FrecuenciaRespiratoria INT, -- Respiraciones  
por minuto

Peso DECIMAL(5,2), -- Kg

Talla DECIMAL(3,2), -- M

ExploracionDetallada NVARCHAR(MAX), --  
Datos adicionales sobre percances en el cuerpo  
del paciente.

-- 6.1.3 Resultados Previos y Actuales

ResultadosEstudios NVARCHAR(MAX), -- De  
laboratorio, gabinete, otros

-- 6.1.4 Diagnosticos

DiagnosticosProblemasClinicos  
NVARCHAR(MAX),

-- 6.1.5 Pronostico

Pronostico NVARCHAR(MAX),

-- 6.1.6 Indicacion Terapeutica

IndicacionTerapeutica NVARCHAR(MAX),

FOREIGN KEY (PacienteID) REFERENCES  
Pacientes(PacienteID),

FOREIGN KEY (EstablecimientoID)  
REFERENCES  
Establecimientos(EstablecimientoID),

FOREIGN KEY (PersonalSaludID)  
REFERENCES PersonalSalud(PersonalID)

);

GO

CREATE TABLE NotasEvolucion (

NotaEvolucionID INT PRIMARY KEY  
IDENTITY(1,1),

PacienteID INT NOT NULL,

EstablecimientoID INT NOT NULL,

PersonalSaludID INT NOT NULL,

FechaHoraCreacion DATETIME2 NOT NULL,

FirmaCreador NVARCHAR(255),

EdadPacienteAlMomento INT,

SexoPacienteAlMomento NVARCHAR(10),

NumeroCamaExpediente NVARCHAR(50),  
  
-- 6.2.1  
  
EvolucionActualizacionCuadroClinico  
NVARCHAR(MAX),

-- 6.2.2  
  
SignosVitales NVARCHAR(MAX),  
  
-- 6.2.3  
  
ResultadosRelevantesEstudiosAuxiliares  
NVARCHAR(MAX),

-- 6.2.4  
  
DiagnosticosProblemasClinicos  
NVARCHAR(MAX),

-- 6.2.5  
  
Pronostico NVARCHAR(MAX),  
  
-- 6.2.6  
  
TratamientoIndicacionesMedicas  
NVARCHAR(MAX), -- Para medicamentos:  
dosis, via, cada cuando se tienen que tomarlos

FOREIGN KEY (PacienteID) REFERENCES  
Pacientes(PacienteID),  
  
FOREIGN KEY (EstablecimientoID)  
REFERENCES  
Establecimientos(EstablecimientoID),

FOREIGN KEY (PersonalSaludID)  
REFERENCES PersonalSalud(PersonalID)  
  
);  
  
GO

-- Tabla para Notas de Interconsulta (6.3)  
  
CREATE TABLE NotasInterconsulta (  
  
    NotaInterconsultaID INT PRIMARY KEY  
    IDENTITY(1,1),  
  
    PacienteID INT NOT NULL,  
  
    EstablecimientoID INT NOT NULL, --  
    Establecimiento donde se genera la consulta  
  
    MedicoSolicitanteID INT NOT NULL, --  
    Medico que solicita la consulta  
  
    MedicoConsultadoID INT NOT NULL, --  
    Medico que realiza la interconsulta y elabora la  
    nota  
  
    FechaHoraCreacion DATETIME2 NOT NULL,  
    -- Esta se usa para cuando el medico  
    especialista hace y guarda su respuesta  
  
    FirmaCreador NVARCHAR(255), -- Firma del  
    medico consultado  
  
    EdadPacienteAlMomento INT,  
  
    SexoPacienteAlMomento NVARCHAR(10),  
  
    NumeroCamaExpediente NVARCHAR(50),  
  
    FechaHoraSolicitud DATETIME2, -- Fecha y  
    hora en que se solicito la interconsulta  
  
    MotivoSolicitud NVARCHAR(MAX), -- Motivo  
    por el cual se solicita la interconsulta

-- Notas del medico consultado (6.3.1, 6.3.2, 6.3.3)

CriteriosDiagnosticos NVARCHAR(MAX),

PlanEstudios NVARCHAR(MAX),

SugerenciasDiagnosticasTratamiento  
NVARCHAR(MAX),

FOREIGN KEY (PacienteID) REFERENCES  
Pacientes(PacienteID),

FOREIGN KEY (EstablecimientoID)  
REFERENCES  
Establecimientos(EstablecimientoID),

FOREIGN KEY (MedicoSolicitanteID)  
REFERENCES PersonalSalud(PersonalID),

FOREIGN KEY (MedicoConsultadoID)  
REFERENCES PersonalSalud(PersonalID)

);

GO

-- Tabla para Notas de Referencia/Traslado (6.4)

CREATE TABLE NotasReferenciaTraslado (

NotaReferenciaTrasladoID INT PRIMARY  
KEY IDENTITY(1,1),

PacienteID INT NOT NULL,

MedicoEnviaID INT NOT NULL,

FechaHoraCreacion DATETIME2 NOT NULL,

FirmaCreador NVARCHAR(255),

EdadPacienteAlMomento INT,

SexoPacienteAlMomento NVARCHAR(10),

NumeroCamaExpediente NVARCHAR(50),

-- 6.4.1

EstablecimientoEnviaID INT NOT NULL, --  
Foreign key a Establecimientos

-- 6.4.2

NombreEstablecimientoReceptor  
NVARCHAR(255) NOT NULL, -- Nombre del  
establecimiento que recibe

DomicilioEstablecimientoReceptor  
NVARCHAR(MAX), -- Domicilio del  
establecimiento receptor

-- 6.4.3

MotivoEnvio NVARCHAR(MAX) NOT NULL,

ImpresionDiagnostica NVARCHAR(MAX)  
NOT NULL,

TerapeuticaEmpleada NVARCHAR(MAX), --  
Si la hubo

FOREIGN KEY (PacienteID) REFERENCES  
Pacientes(PacienteID),

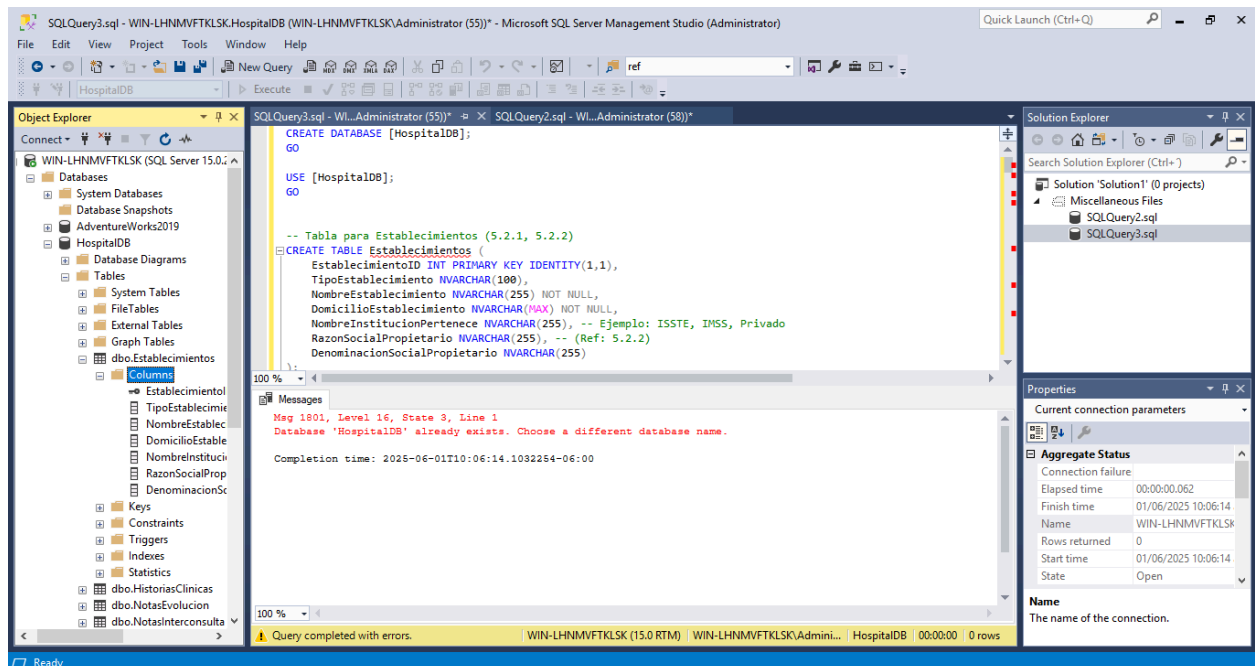
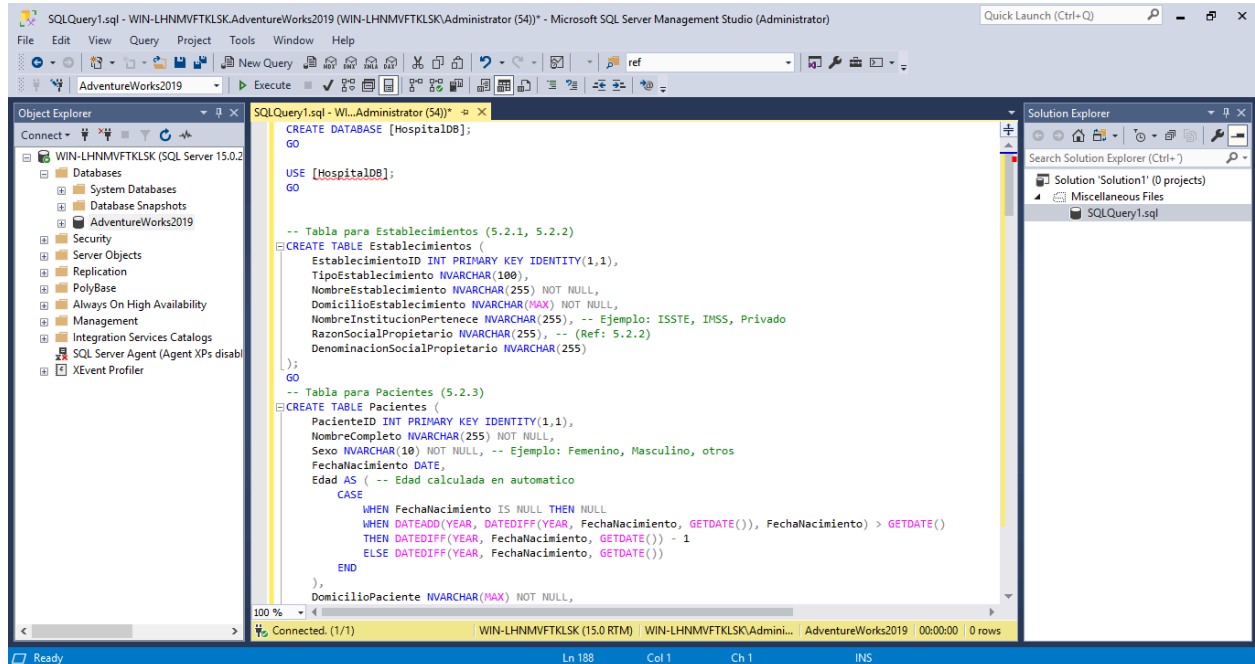
FOREIGN KEY (EstablecimientoEnviaID)  
REFERENCES  
Establecimientos(EstablecimientoID),

FOREIGN KEY (MedicoEnviaID)  
REFERENCES PersonalSalud(PersonalID)

);

GO

## Resultados Obtenidos:



Nota de autor: No estoy seguro como o exactamente que paso, pero al momento de crear la tabla tuve múltiples errores donde la base intentaba crear una ya de “preset” o no se guardaba en la carpeta a la izquierda de “Databases” al final si se logro pero al momento de tratar de ejecutar marcaba error de que ya existian. De igual forma después de una breve revisión, se noto que de igual forma las columnas si se habían creado asi que se decidió de igual forma proseguir

## 2.- Crear inicios de sesión, usuarios, roles y asignación de permisos

Se implementaron configuraciones de seguridad básicas para la base de datos, incluyendo la creación de un nuevo inicio de sesión, un usuario de base de datos con permisos de solo lectura, y un rol personalizado con permisos específicos sobre una tabla.

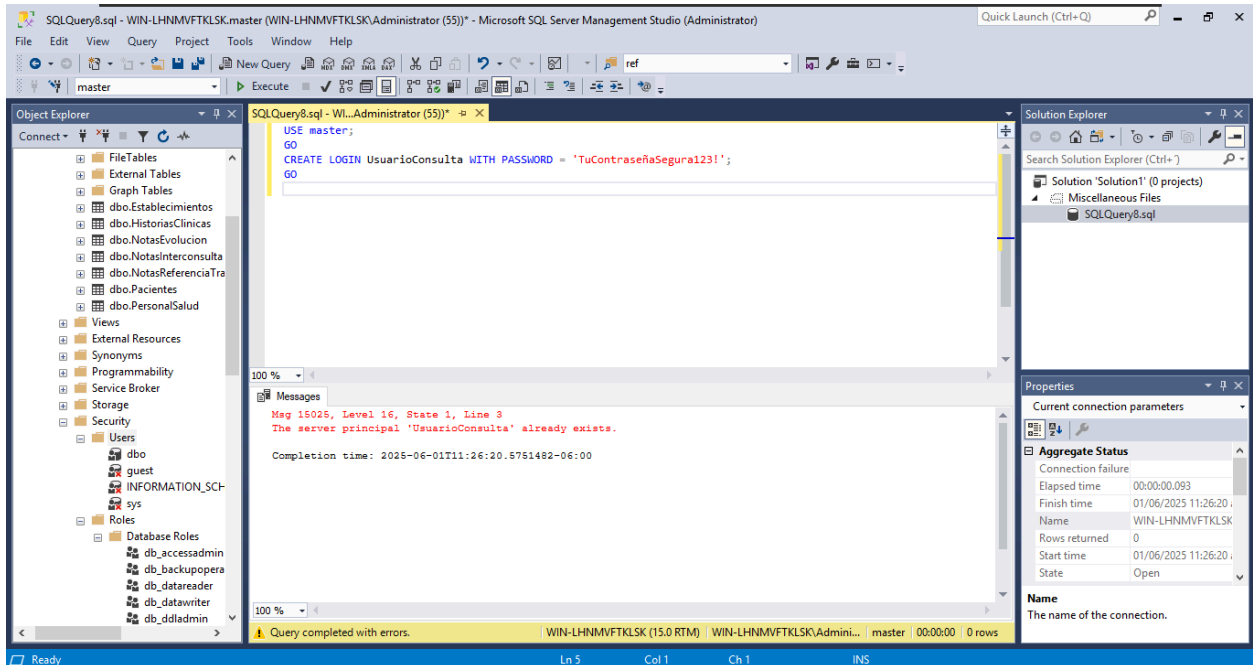
### Script SQL:

- **Creación de nuevo login:**  
USE master;  
GO  
CREATE LOGIN UsuarioConsulta WITH PASSWORD = 'TuContraseñaSegura123!';  
GO
- **Creación de usuario:**  
USE HospitalDB;  
GO  
CREATE USER UsuarioConsultaHospital FROM LOGIN UsuarioConsulta;  
GO  
-- Rol de solo lectura (db\_datareader) para el usuario  
ALTER ROLE db\_datareader ADD MEMBER UsuarioConsultaHospital;  
GO  
Nota: el rol de db\_datareader es un rol generado automáticamente por SSMS para que el usuario solo lea la base de datos. Se le pueden asignar más roles para
- **Creación de un rol un rol y asignarle permisos de SELECT sobre una sola tabla**  
USE [HospitalDB]  
GO  
CREATE ROLE [RolVisualizadorPacientes]  
GO  
GRANT SELECT ON dbo.Pacientes TO RolVisualizadorPacientes;  
GO

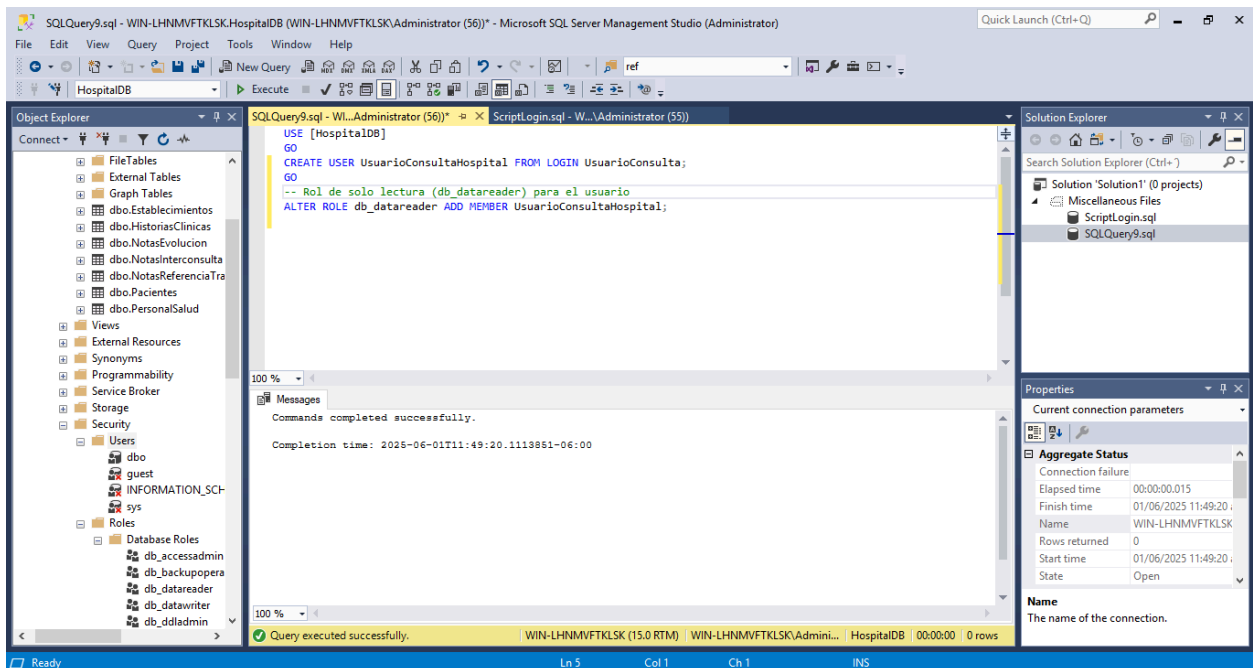
### Capturas:

- **Creación de nuevo inicio de sesión (UsuarioConsulta):**

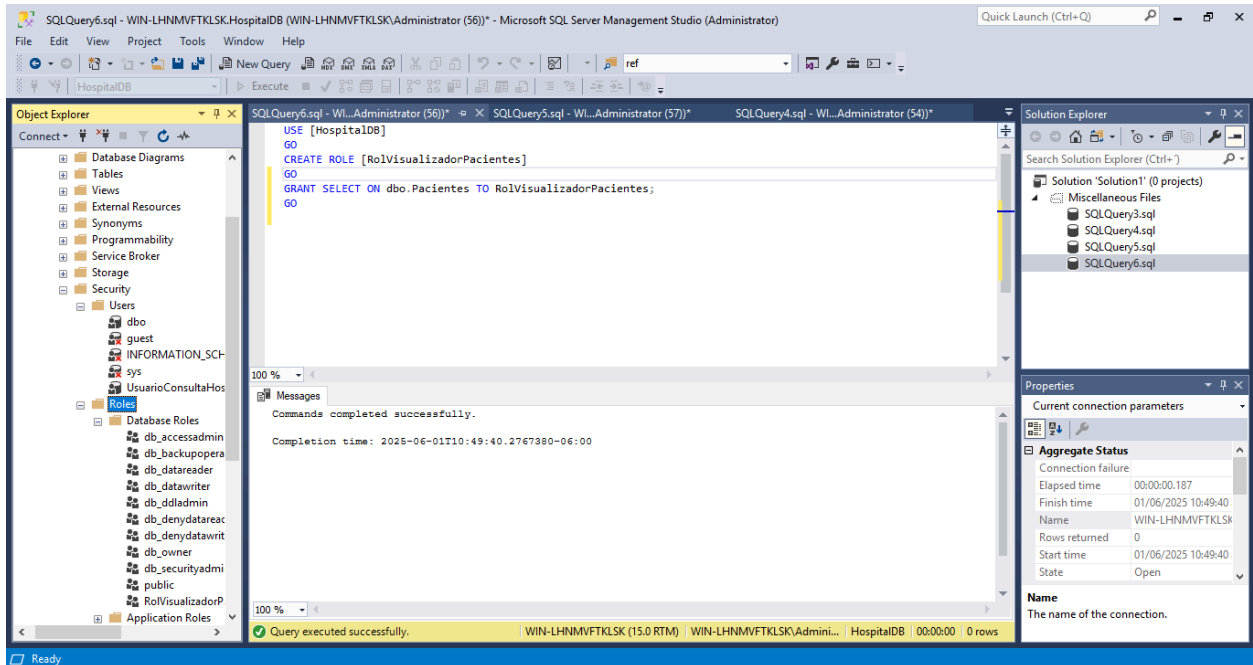




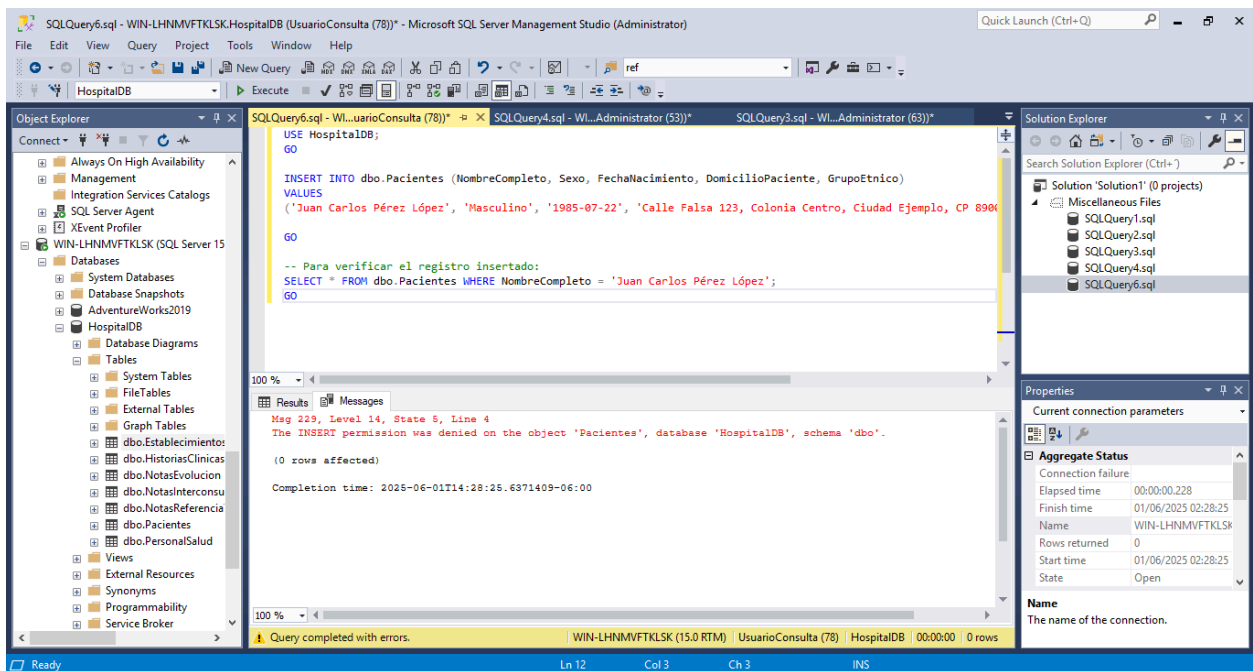
- **Creación de usuario de solo lectura:**



- **Creación de rol personalizado (RolVisualizadorPacientes) y asignación de permisos SELECT sobre la tabla PACIENTE:**



- **Verificación de comportamiento con permisos:**
  - No puede insertar datos



- Si puede ver los datos gracias a los permisos dados por (db\_datareader)

IntentoFallidoDeInsertDeUsuario.sql - WIN-LHNMVFTKLSK.HospitalDB (UsuarioConsulta (78)) - Microsoft SQL Server Management Studio (Administrator)

```

USE HospitalDB;
GO

SELECT * FROM dbo.NotasEvolucion;
GO

```

PacienteID	NombreCompleto	Sexo	FechaNacimiento	Edad	DomicilioPaciente	GrupoEtni
1	Maria Guadalupe Rodríguez Sánchez	Femenino	1992-03-15	33	Avenida Siempre Viva 742, Lomas Verdes, Ciudad E...	Indígena
2	Pedro Antonio García Martínez	Masculino	1978-11-01	46	Boulevard de los Sueños 45, El Carmen, Villa Hemo...	Mestizo
3	Ana Sofía González Ramírez	Femenino	2001-09-28	23	Privada del Sol 210, Residencial del Bosque, Puerto...	Blanco
4	Luis Fernando Díaz Castro	Masculino	1965-05-10	60	Calle de la Luna 30, Barrio Antiguo, Ciudad Colonial...	Afrodesc
5	Laura Elena Vargas Soto	Femenino	1989-01-05	36	Cerrada de la Paz 15, Jardines del Valle, Ciudad Tra...	Mestizo
6	Roberto Carlos Tones Morales	Masculino	1995-08-19	29	Andador Estrella 88, Colonia Moderna, Nuevo Horiz...	Asiático
7	Sofía Alejandra Jiménez Pineda	Femenino	1973-02-25	52	Ciruito del Lago 112, Fraccionamiento Las Fuentes...	Indígena
8	Diego Armando Ruiz Flores	Masculino	2005-04-03	20	Avenida Principal 50, Centro Histórico, Villa Vieja...	Mestizo
9	Gabriela Fernanda Herrera Cruz	Femenino	1980-12-12	44	Prolongación Olivo 7, Sector Palmas, Costa Azul, CP...	Blanco
10	Miguel Ángel Solís Navaro	Masculino	1960-06-07	64	Callesón del Mar 22, Zona Costera, Puerto Dulce, CP...	Mestizo

Query executed successfully. WIN-LHNMVFTKLSK (15.0 RTM) | UsuarioConsulta (78) | HospitalDB | 00:00:00 | 10 rows

En cuanto a la pregunta de: “¿Cómo se podrían revocar estos permisos?” Para revocar permisos en SQL Server, se utiliza el comando REVOKE o también usar SSMS para poder brindar o quitar permisos a un usuario. Procederemos a hacerlo de ambas formas:

- **Asignacion de rol (RolVisualizadorPacientes):**

SQLQuery16.sql - WIN-LHNMVFTKLSK.HospitalDB (WIN-LHNMVFTKLSK\Administrator (59)) - Microsoft SQL Server Management Studio (Administrator)

```

USE HospitalDB;
GO

ALTER ROLE [RolVisualizadorPacientes] ADD MEMBER UsuarioConsultaHospital;
GO

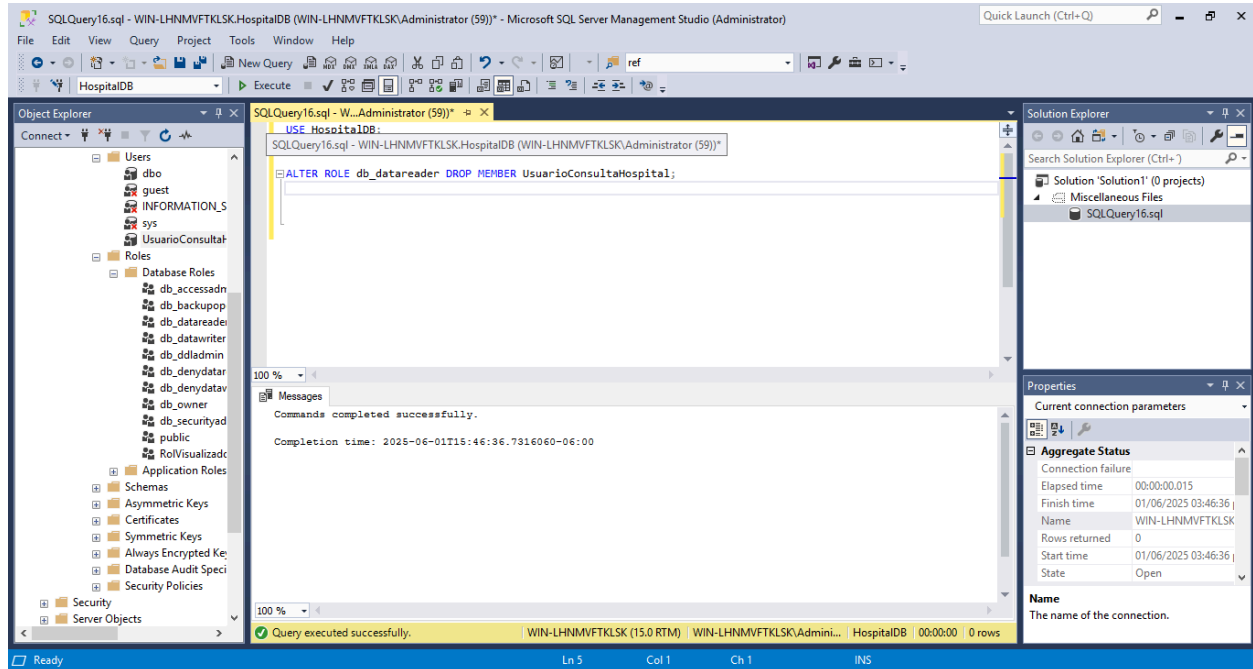
```

PacienteID	NombreCompleto	Sexo	FechaNacimiento	Edad	DomicilioPaciente	GrupoEtni
1	Maria Guadalupe Rodríguez Sánchez	Femenino	1992-03-15	33	Avenida Siempre Viva 742, Lomas Verdes, Ciudad E...	Indígena
2	Pedro Antonio García Martínez	Masculino	1978-11-01	46	Boulevard de los Sueños 45, El Carmen, Villa Hemo...	Mestizo
3	Ana Sofía González Ramírez	Femenino	2001-09-28	23	Privada del Sol 210, Residencial del Bosque, Puerto...	Blanco
4	Luis Fernando Díaz Castro	Masculino	1965-05-10	60	Calle de la Luna 30, Barrio Antiguo, Ciudad Colonial...	Afrodesc
5	Laura Elena Vargas Soto	Femenino	1989-01-05	36	Cerrada de la Paz 15, Jardines del Valle, Ciudad Tra...	Mestizo
6	Roberto Carlos Tones Morales	Masculino	1995-08-19	29	Andador Estrella 88, Colonia Moderna, Nuevo Horiz...	Asiático
7	Sofía Alejandra Jiménez Pineda	Femenino	1973-02-25	52	Ciruito del Lago 112, Fraccionamiento Las Fuentes...	Indígena
8	Diego Armando Ruiz Flores	Masculino	2005-04-03	20	Avenida Principal 50, Centro Histórico, Villa Vieja...	Mestizo
9	Gabriela Fernanda Herrera Cruz	Femenino	1980-12-12	44	Prolongación Olivo 7, Sector Palmas, Costa Azul, CP...	Blanco
10	Miguel Ángel Solís Navaro	Masculino	1960-06-07	64	Callesón del Mar 22, Zona Costera, Puerto Dulce, CP...	Mestizo

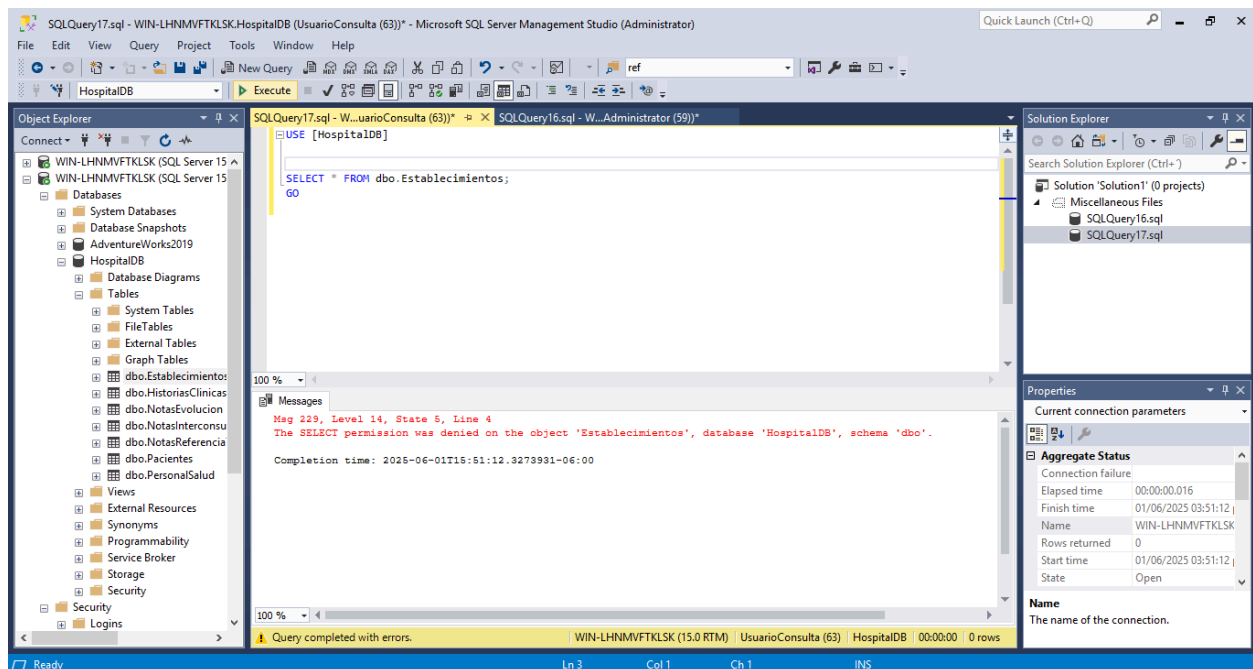
Commands completed successfully. Completion time: 2025-06-01T15:42:49.0135591-06:00

Query executed successfully. WIN-LHNMVFTKLSK (15.0 RTM) | WIN-LHNMVFTKLSK\Admini... | HospitalDB | 00:00:00 | 0 rows

- **Revocacion de rol (db\_datareader):**



- **Verificación de permisos (no puede ver otra tabla que no sea Paciente):**



SQLQuery17.sql - WIN-LHNMVFTKLSK.HospitalDB (UsuarioConsulta (63)) - Microsoft SQL Server Management Studio (Administrator)

USE [HospitalDB]

```
SELECT * FROM dbo.Pacientes;
```

Results

PacienteID	NombreCompleto	Sexo	FechaNacimiento	Edad	DomicilioPaciente	GrupoEtni
1	Maria Guadalupe Rodríguez Sánchez	Femenino	1992-03-15	33	Avenida Siempre Viva 742, Lomas Verdes, Ciudad E...	Indígena
2	Pedro Antonio García Martínez	Masculino	1978-11-01	46	Boulevard de los Sueños 45, El Carmen, Villa Hemo...	Mestizo
3	Ana Sofía González Ramírez	Femenino	2001-09-28	23	Privada del Sol 210, Residencial del Bosque, Puerto...	Blanco
4	Luis Fernando Díaz Castro	Masculino	1965-05-10	60	Calle de la Luna 30, Barrio Antiguo, Ciudad Colonial, ...	Afrodesce
5	Laura Elena Vargas Soto	Femenino	1989-01-05	36	Cerrada de la Paz 15, Jardines del Valle, Ciudad Tra...	Mestizo
6	Roberto Carlos Torres Morales	Masculino	1995-08-19	29	Andador Estrella 88, Colonia Moderna, Nuevo Horiz...	Asiático
7	Sofía Alejandra Jiménez Pineda	Femenino	1973-02-25	52	Circo del Lago 112, Fraccionamiento Las Fuentes...	Indígena
8	Diego Armando Ruiz Flores	Masculino	2005-04-03	20	Avenida Principal 50, Centro Histórico, Villa Vieja, C...	Mestizo
9	Gabriela Fernanda Herrera Cruz	Femenino	1980-12-12	44	Prolongación Olivo 7, Sector Palmas, Costa Azul, CP...	Blanco
10	Miguel Ángel Solís Navaro	Masculino	1960-06-07	64	Callesón del Mar 22, Zona Costera, Puerto Dulce, CP...	Mestizo

Query executed successfully. WIN-LHNMVFTKLSK (15.0 RTM) UsuarioConsulta (63) HospitalDB 00:00:00 10 rows

Ahora haremos lo inverso pero con SSMS:

- Se desmarca la casilla donde esta el Rol (RolVisualizadorPacientes) y marca la casilla (db\_datareader)

SQLQuery16.sql - WIN-LHNMVFTKLSK.HospitalDB (WIN-LHNMVFTKLSK\Administrador (59)) - Microsoft SQL Server Management Studio (Administrator)

Database User - UsuarioConsultaHospital

Select a page: General, Owned Schemas, Membership, Securables, Extended Properties

Database role membership:

Role Members

- ☐ db\_accessadmin
- ☐ db\_backupoperator
- ☒ db\_datareader
- ☐ db\_datawriter
- ☐ db\_doadmin
- ☐ db\_denydatareader
- ☐ db\_denydatawriter
- ☐ db\_owner
- ☐ db\_securityadmin
- ☐ RolVisualizadorPacientes

Connection

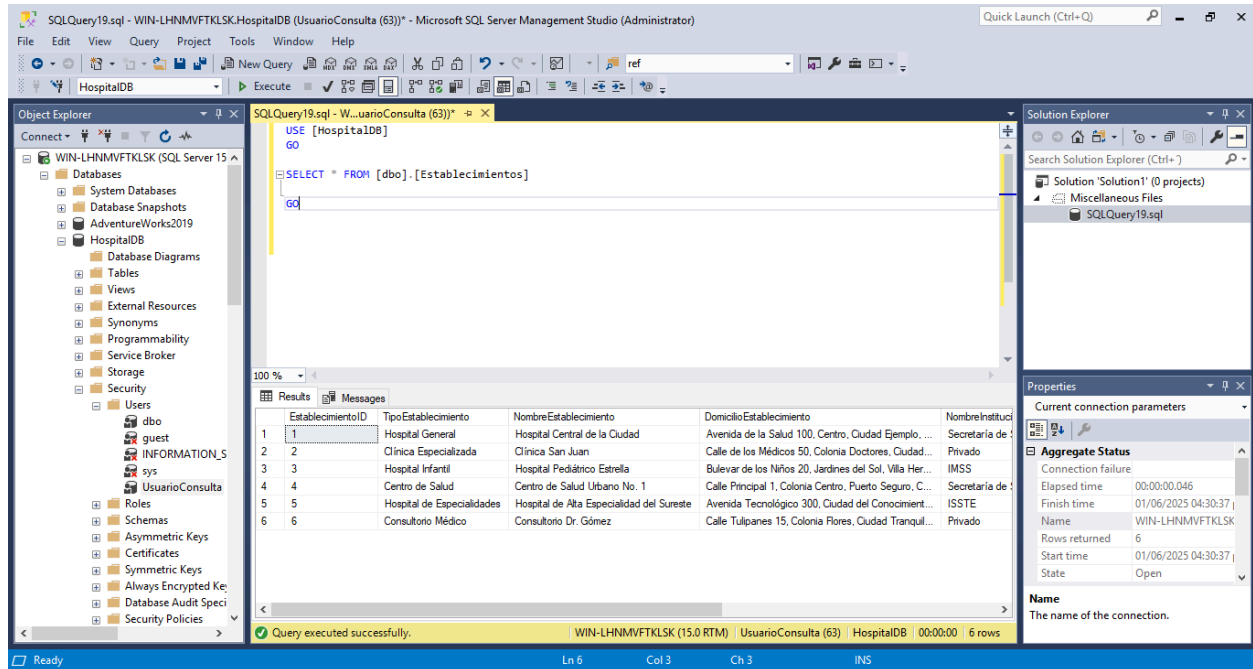
Server: WIN-LHNMVFTKLSK

Connection: WIN-LHNMVFTKLSK\Administrador

Progress

Ready

Query executed successfully. WIN-LHNMVFTKLSK (15.0 RTM) UsuarioConsulta (63) HospitalDB 00:00:00 0 rows

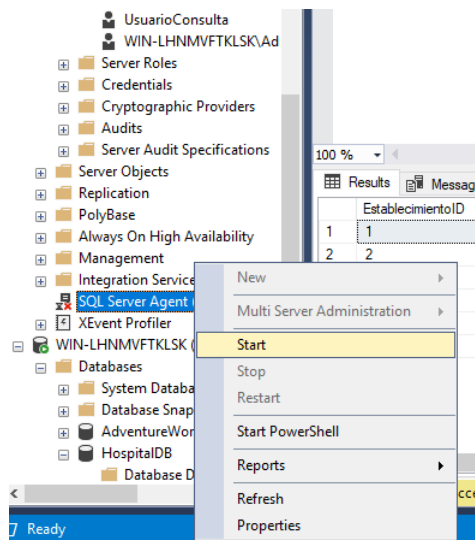


Nota: Hubo algunas complicaciones con que se resolvieron reiniciando las conexiones.

### 3.- Crear y programar tareas automáticas en SQL Server

Se hará la configuración de una tarea automática utilizando SSMS para realizar respaldos completos diarios de la base de datos *HospitalDB*. Esto asegurará la disponibilidad de copias de seguridad regulares para la recuperación de datos en caso de ser necesario.

- **Activación de SQL Server Agent:**
  - Simplemente se busca en el recuadro de “Object Explorer” y se le da a **“Start”**



- **Script SQL para Creación y Programación del Job de Respaldo:**

```
USE msdb; -- Los Jobs se configuran en la base de datos msdb
GO
```

```
DECLARE @JobName NVARCHAR(128) = N'Respaldo Diario HospitalDB';
DECLARE @BackupPath NVARCHAR(260) = N'C:\Program Files\Microsoft SQL
Server\MSSQL15.MSSQLSERVER\MSSQL\Backup\'; -- CAMBIA ruta si quieres guardar el respaldo en
otro disco o ruta de red
```

```
DECLARE @DatabaseName NVARCHAR(128) = N'HospitalDB';
DECLARE @JobEnabled BIT = 1; -- 1 = habilitado, 0 = deshabilitado
```

```
DECLARE @ScheduleName NVARCHAR(128) = N'HorarioRespaldoDiarioHospitalDB';
DECLARE @StepName NVARCHAR(128) = N'Ejecutar Respaldo HospitalDB';
DECLARE @OwnerLoginName NVARCHAR(128) = N'sa'; -- Cambiar si quieres que otro login administre
el job
DECLARE @BackupCommand NVARCHAR(MAX);
DECLARE @ScheduleId INT;
```

```
-- Eliminar Job y Schedule anterior (si existen), para evitar duplicados al volver a ejecutar el script
IF EXISTS (SELECT job_id FROM dbo.sysjobs_view WHERE name = @JobName)
BEGIN
    EXEC dbo.sp_delete_job @job_name = @JobName, @delete_unused_schedule = 1;
END
```

```
IF EXISTS (SELECT schedule_id FROM dbo.sysschedules WHERE name = @ScheduleName)
BEGIN
    SELECT TOP 1 @ScheduleId = schedule_id FROM dbo.sysschedules WHERE name =
@ScheduleName;
    IF NOT EXISTS (
        SELECT 1 FROM dbo.sysjobschedules
        WHERE schedule_id = @ScheduleId
        AND job_id <> (SELECT job_id FROM dbo.sysjobs WHERE name = @JobName)
    )
    BEGIN
        BEGIN TRY
            IF @ScheduleId IS NOT NULL
            BEGIN
                EXEC dbo.sp_detach_schedule @job_name = @JobName, @schedule_id = @ScheduleId,
@delete_automatic_del_schedule = 0;
                END
                EXEC dbo.sp_delete_schedule @schedule_name = @ScheduleName, @force_delete = 1;
            END TRY
            BEGIN CATCH
                -- Se ignora si no se puede eliminar el schedule
            END CATCH
        END
    END
END
```

```

-- Crear Job
EXEC dbo.sp_add_job
    @job_name = @JobName,
    @enabled = @JobEnabled,
    @description = N'Realiza un respaldo completo diario de la base de datos HospitalDB.',
    @owner_login_name = @OwnerLoginName,
    @notify_level_email = 0,
    @notify_level_netsend = 0,
    @notify_level_page = 0,
    @delete_level = 0;

-- Esto al ejecutarse el respaldo lo crea con un nombre dinamico basado en fecha y hora
SET @BackupCommand = N'DECLARE @BackupFileName NVARCHAR(500);' + CHAR(13) +
CHAR(10) +
    N'SET @BackupFileName = N''' + @BackupPath + @DatabaseName +
    N'_FullBackup_' + REPLACE(CONVERT(VARCHAR(10), GETDATE(), 112), "-", "") + "_" + '
+
    N'REPLACE(CONVERT(VARCHAR(8), GETDATE(), 108), ":", "") + ".bak";' + CHAR(13) +
CHAR(10) +
    N'BACKUP DATABASE [' + @DatabaseName + N'] ' + CHAR(13) + CHAR(10) +
    N'TO DISK = @BackupFileName ' + CHAR(13) + CHAR(10) +
    N'WITH NOFORMAT, INIT, NAME = N''' + @DatabaseName + N'-Full Database Backup', ' +
CHAR(13) + CHAR(10) +
    N'SKIP, NOREWIND, NOUNLOAD, STATS = 10;';

EXEC dbo.sp_add_jobstep
    @job_name = @JobName,
    @step_name = @StepName,
    @step_id = 1,
    @cmdexec_success_code = 0,
    @on_success_action = 1, -- 1 = continuar al siguiente paso o finalizar si es el ultimo
    @on_fail_action = 2, -- 2 = detener job al fallar
    @retry_attempts = 0,
    @retry_interval = 0,
    @os_run_priority = 0,
    @subsystem = N'TSQL',
    @command = @BackupCommand,
    @database_name = N'master',
    @flags = 0;

-- Crear el horario
EXEC dbo.sp_add_schedule
    @schedule_name = @ScheduleName,
    @enabled = 1,
    @freq_type = 4, -- Frecuencia del job:
        -- 1 = Una sola vez
        -- 4 = Diario
        -- 8 = Semanal

```



```

-- 16 = Mensual
@freq_interval = 1, -- Ejecutar cada 1 dia (para @freq_type = 4)
@freq_subday_type = 1, -- 1 = Una vez al dia (puedes usar 4 = cada X minutos, o 8 = cada X horas si
quieres multiples veces)
@freq_subday_interval = 0,
@freq_relative_interval = 0,
@freq_recurrence_factor = 1, -- Ejecutar cada X dias/semanas/etc (valor adicional segun freq_type)
@active_start_date = NULL, -- NULL = empieza hoy
@active_end_date = 99991231, -- Fecha lejana para que no expire
@active_start_time = 020000, -- Hora de inicio (02:00 AM = 020000, formato HHMMSS)
@active_end_time = 235959; -- Hora maxima en la que puede ejecutarse

-- Asociar el horario al Job
SET @ScheduleId = NULL;
SELECT @ScheduleId = schedule_id FROM dbo.sysschedules WHERE name = @ScheduleName;

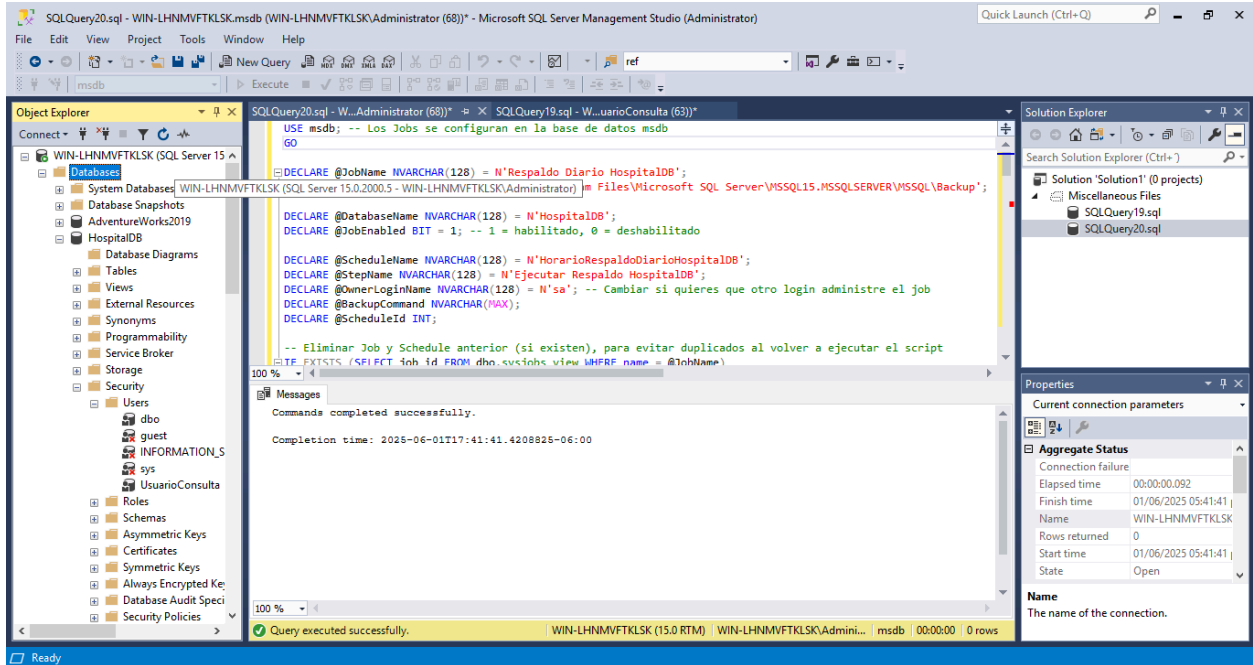
IF @ScheduleId IS NOT NULL
BEGIN
    EXEC dbo.sp_attach_schedule
        @job_name = @JobName,
        @schedule_id = @ScheduleId;
END

-- Establecer el servidor donde se ejecutara el Job
EXEC dbo.sp_add_jobserver
    @job_name = @JobName,
    @server_name = N'(local)'; -- INVESTIGAR Cambiar "(local)" por el nombre de otro servidor si se va a
ejecutar remotamente (cosa de ambientes distribuidos INVESTIGAR)
GO

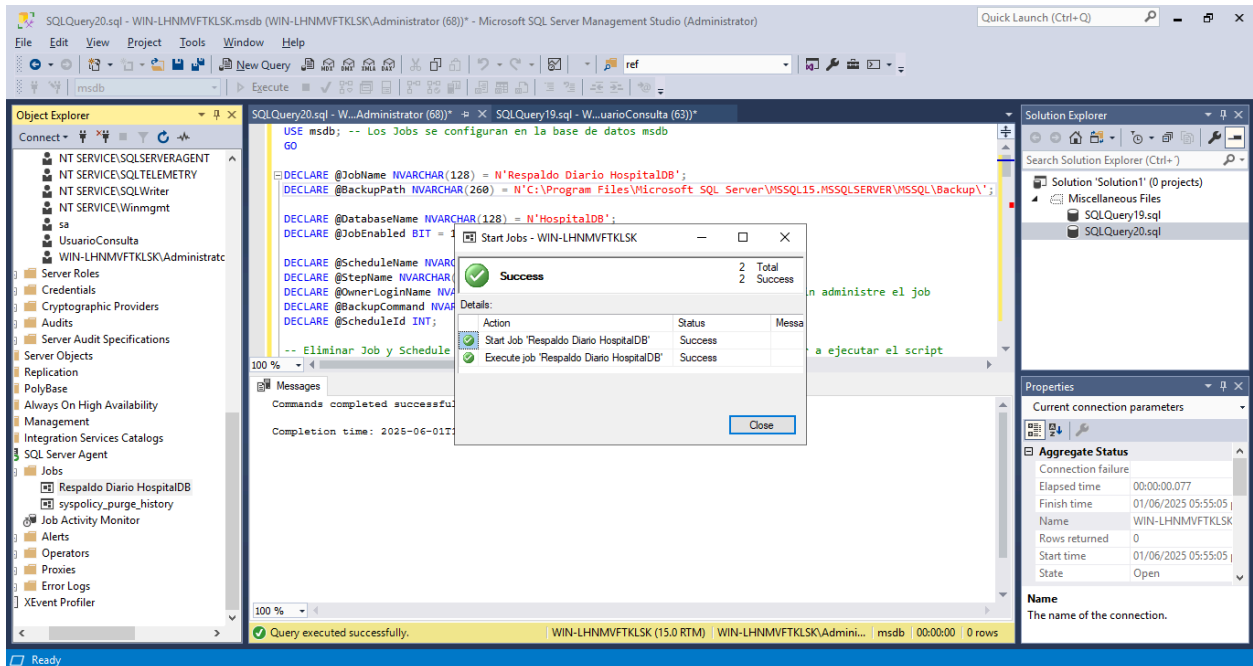
```

## Proceso y Resultados:

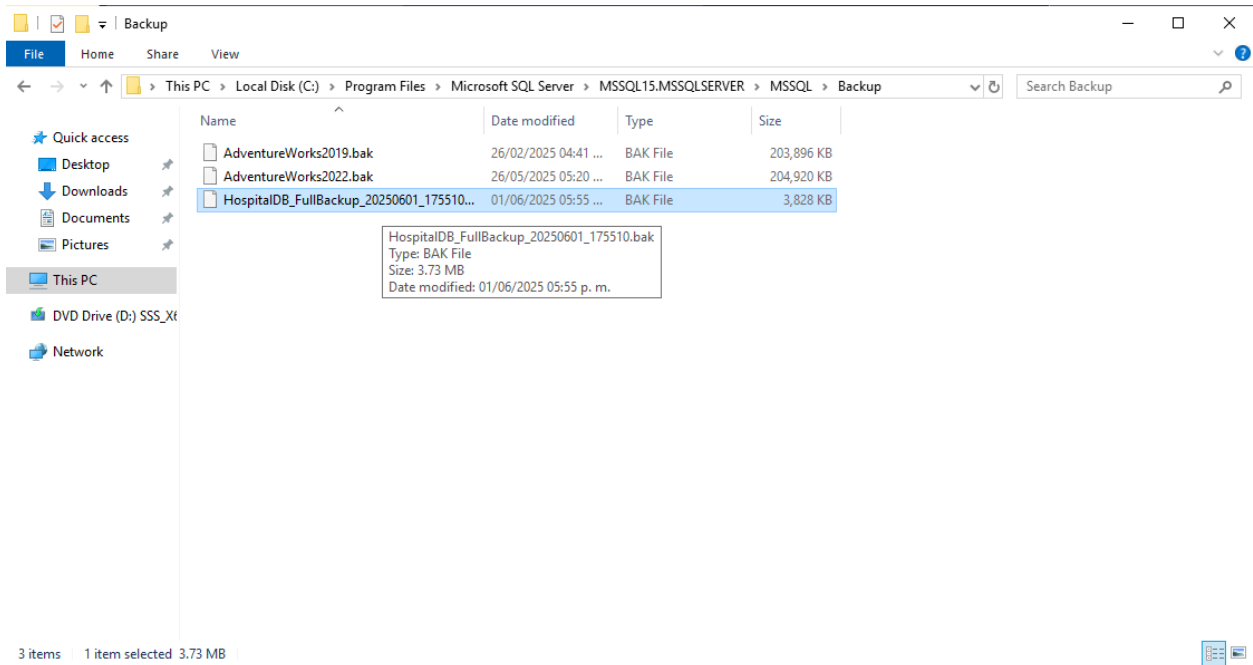
- **Creación del Job de respaldo:**



- **Lanzamiento y prueba del Job:**



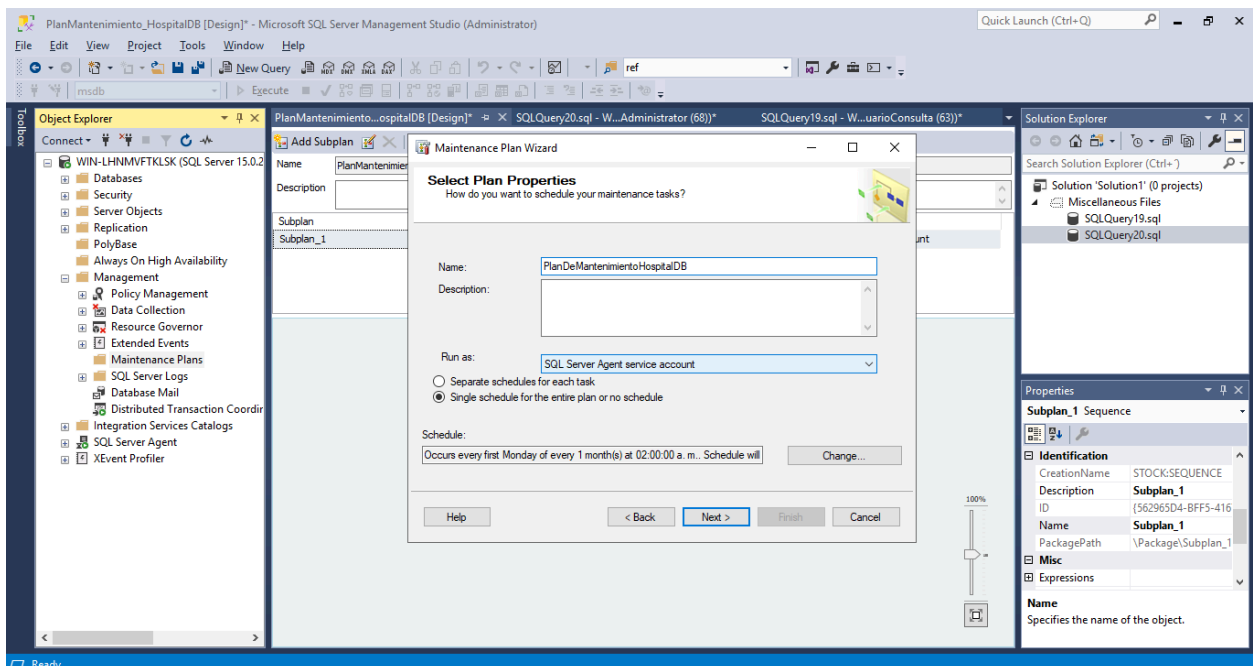
- **Archivo de respaldo generado:**



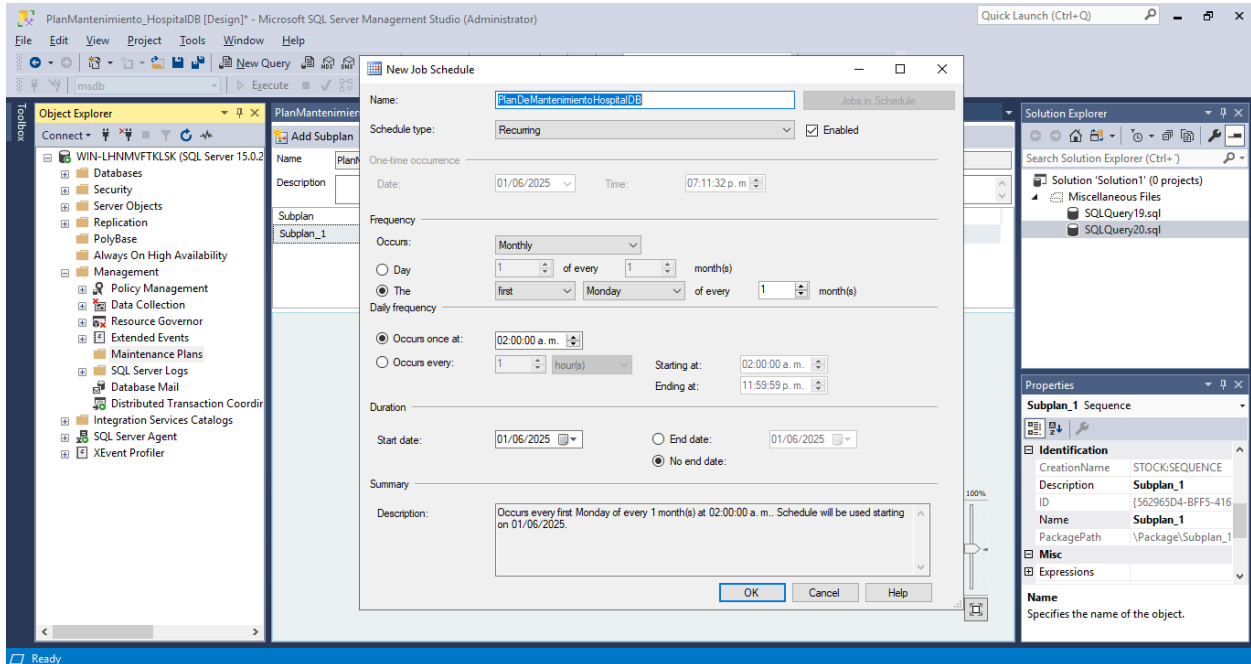
## 4.- Crear un nuevo Plan de Mantenimiento

Se utilizó el Asistente para Planes de Mantenimiento de SSMS (Wizard) para crear un plan que incluye tareas de respaldo completo, reorganización de índices y actualización de estadísticas para la base de datos. Este plan se programó para ejecución periódica:

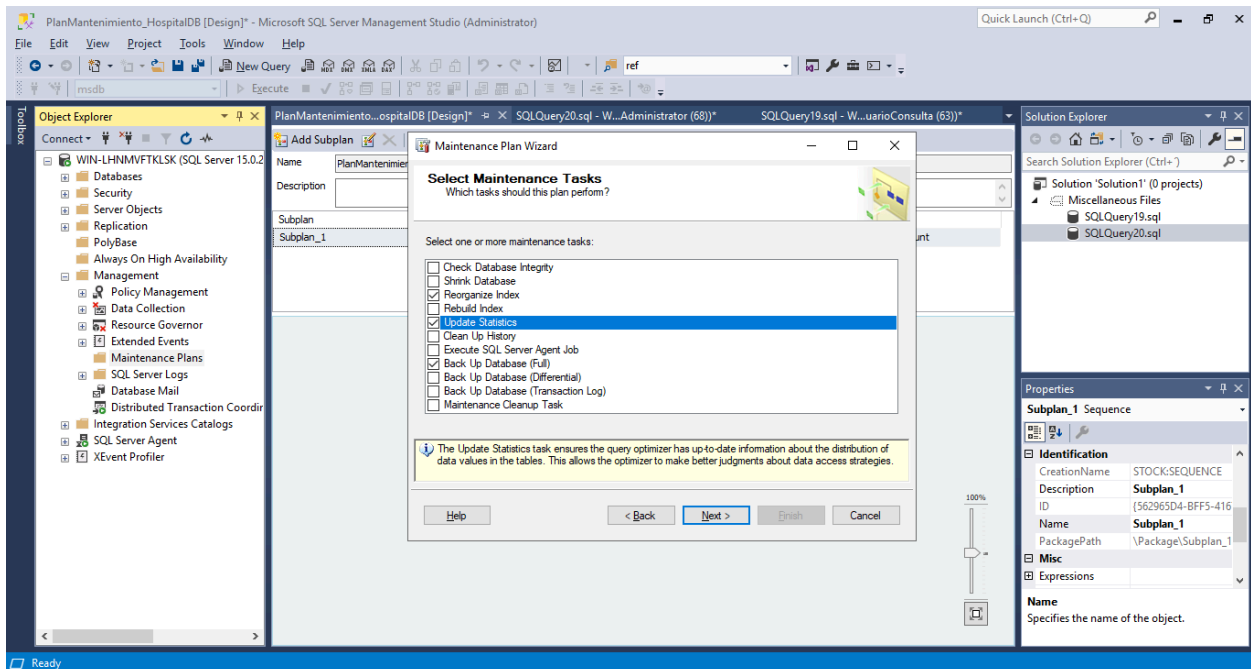
- Creación de plan de Mantenimiento con Nombre “PlanMantenimiento HospitalDB”



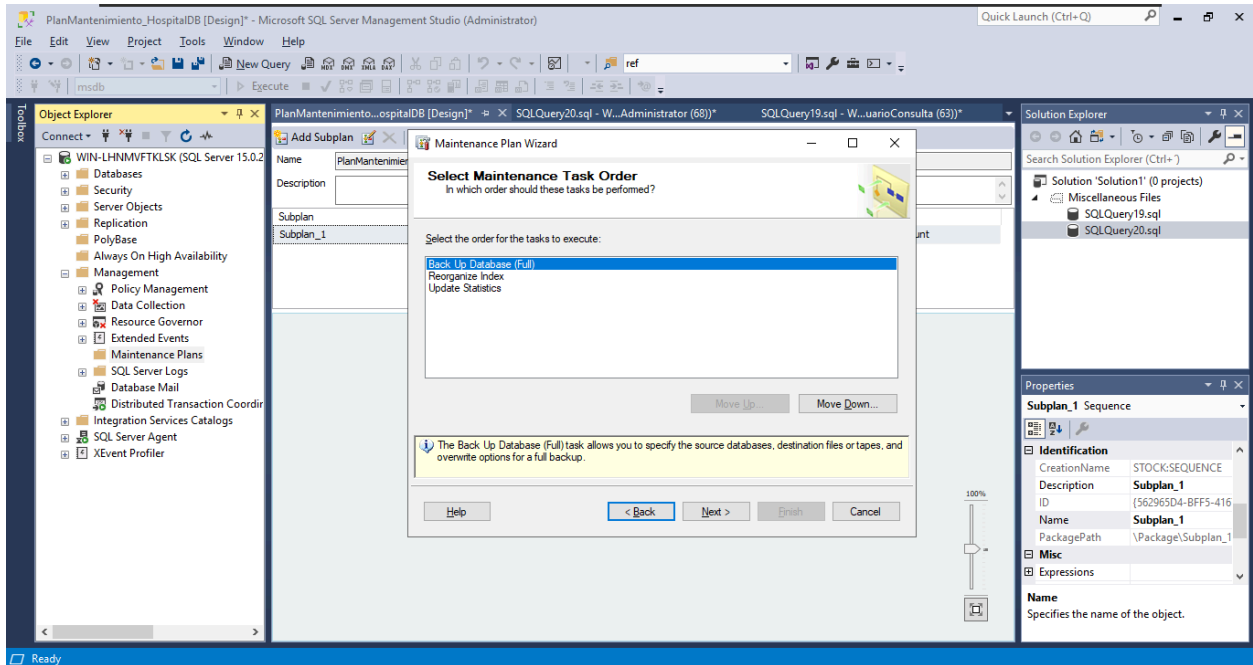
- Se le da el horario de y día de cuando es el respaldo:



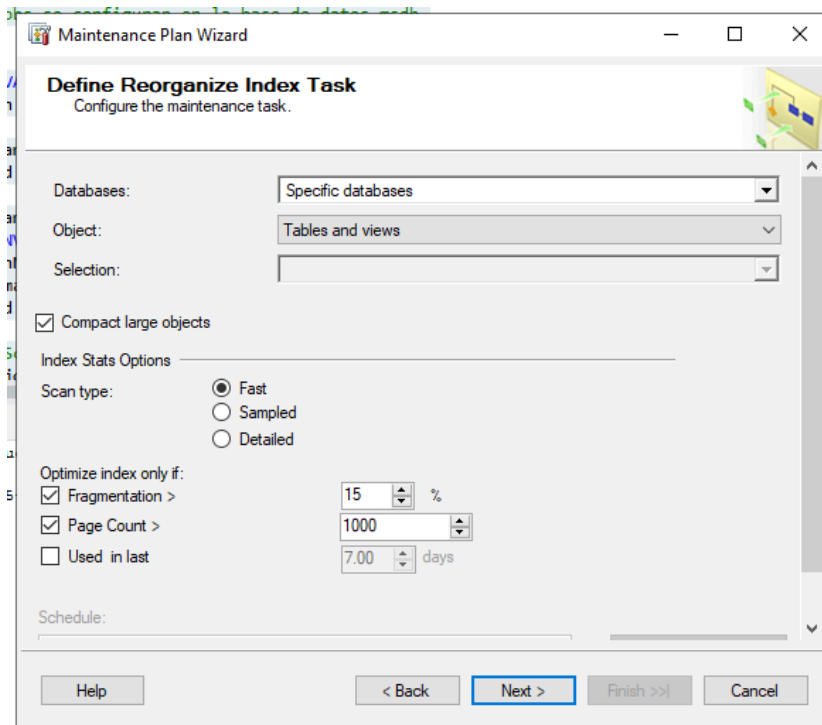
- Se le asignan las 3 tareas que se indicaron (Backup completo, reorganización de índices, actualización de estadísticas):



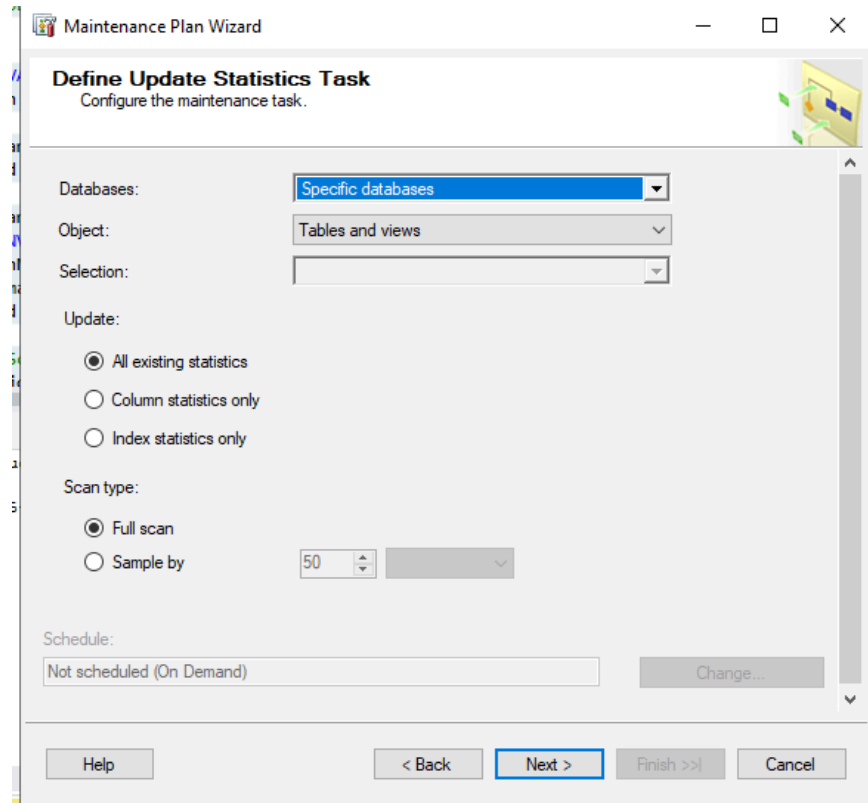
- En este paso damos el orden en el cual se ejecutarán las acciones de mantenimiento:



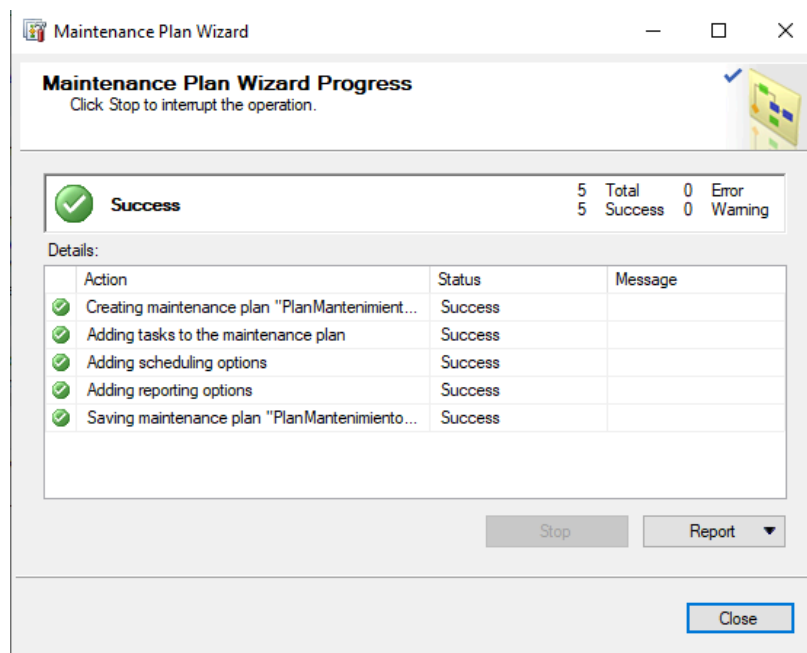
- Se realiza la configuración de la reorganización de índices:



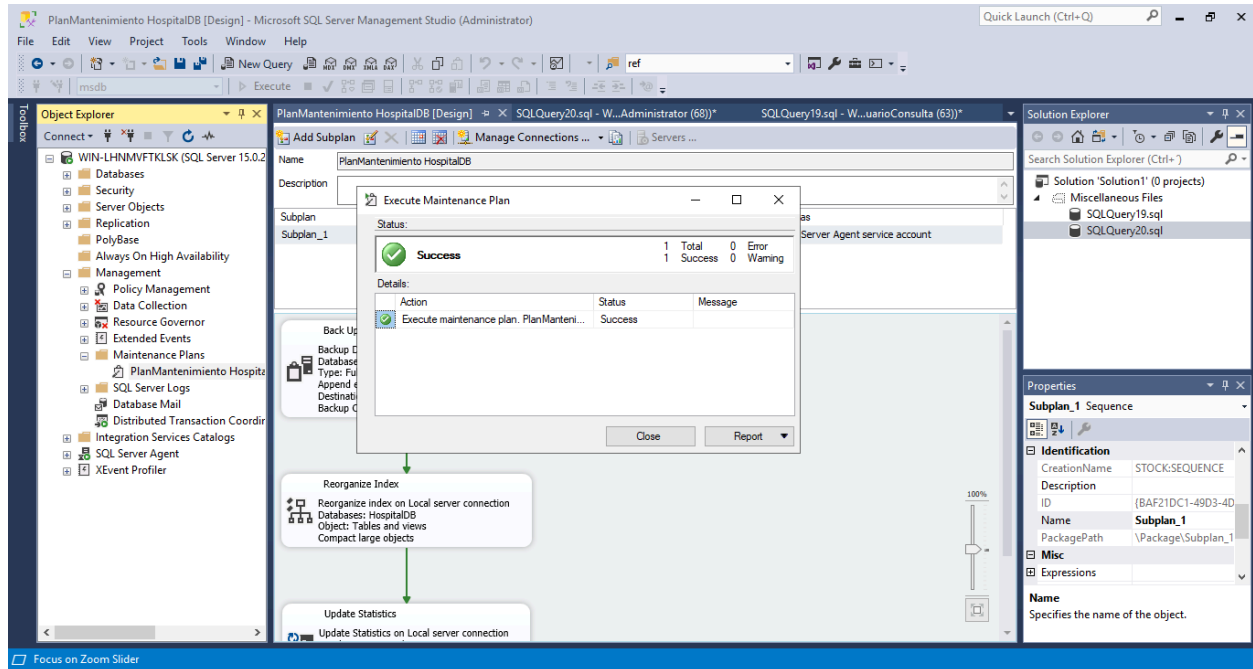
- Se configura la actualización de estadísticas:



- Se ejecuta el Asistente para Planes de Mantenimiento para realizarlo:



- Se lanza por primera vez el plan de mantenimiento:



## 5.- Modificar la base de datos

Se modificó la estructura de todas las tablas relevantes de la base de datos *HospitalDB* para incluir una nueva columna booleana denominada *deleted*. Esta columna se configuró para no permitir valores nulos y tener un valor predeterminado de false (0), facilitando la implementación de borrado lógico.

### Script SQL:

```
USE HospitalDB;
```

```
GO
```

```
ALTER TABLE dbo.Establecimientos
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_Establecimientos_deleted DEFAULT 0;
```

```
GO
```

```
ALTER TABLE dbo.Pacientes
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_Pacientes_deleted DEFAULT 0;
```

```
GO
```

```
ALTER TABLE dbo.PersonalSalud
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_PersonalSalud_deleted DEFAULT 0;
```

```
GO
```

```
ALTER TABLE dbo.HistoriasClinicas
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_HistoriasClinicas_deleted DEFAULT 0;
```

```
GO
```

```
ALTER TABLE dbo.NotasEvolucion
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_NotasEvolucion_deleted DEFAULT 0;
```

```
GO
```

```
ALTER TABLE dbo.NotasInterconsulta
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_NotasInterconsulta_deleted DEFAULT 0;
```

```
GO
```

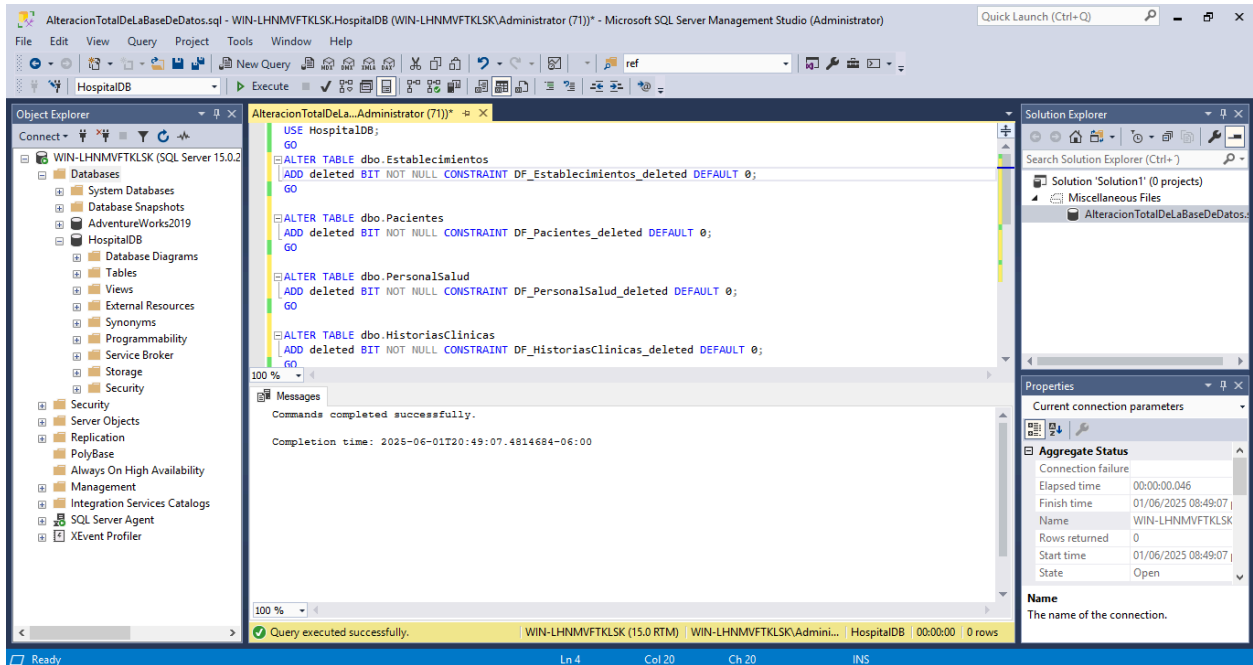
```
ALTER TABLE dbo.NotasReferenciaTraslado
```

```
ADD deleted BIT NOT NULL CONSTRAINT DF_NotasReferenciaTraslado_deleted DEFAULT 0;
```

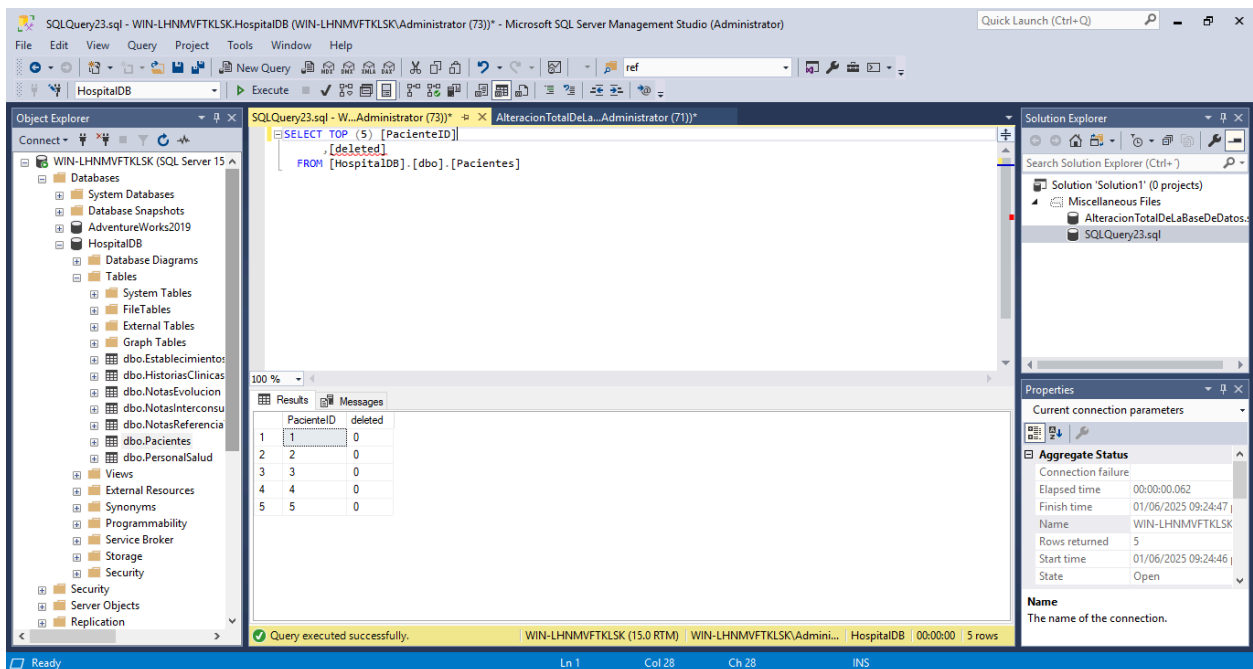
```
GO
```

- **Ejecución exitosa del Script SQL:**

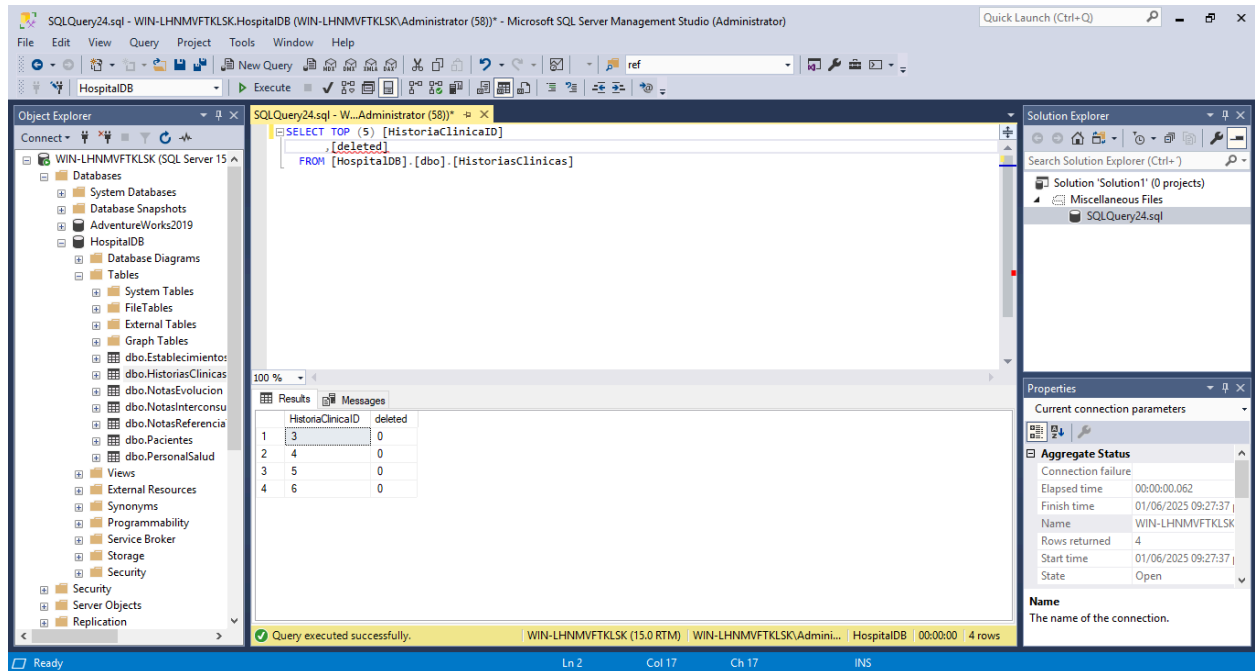




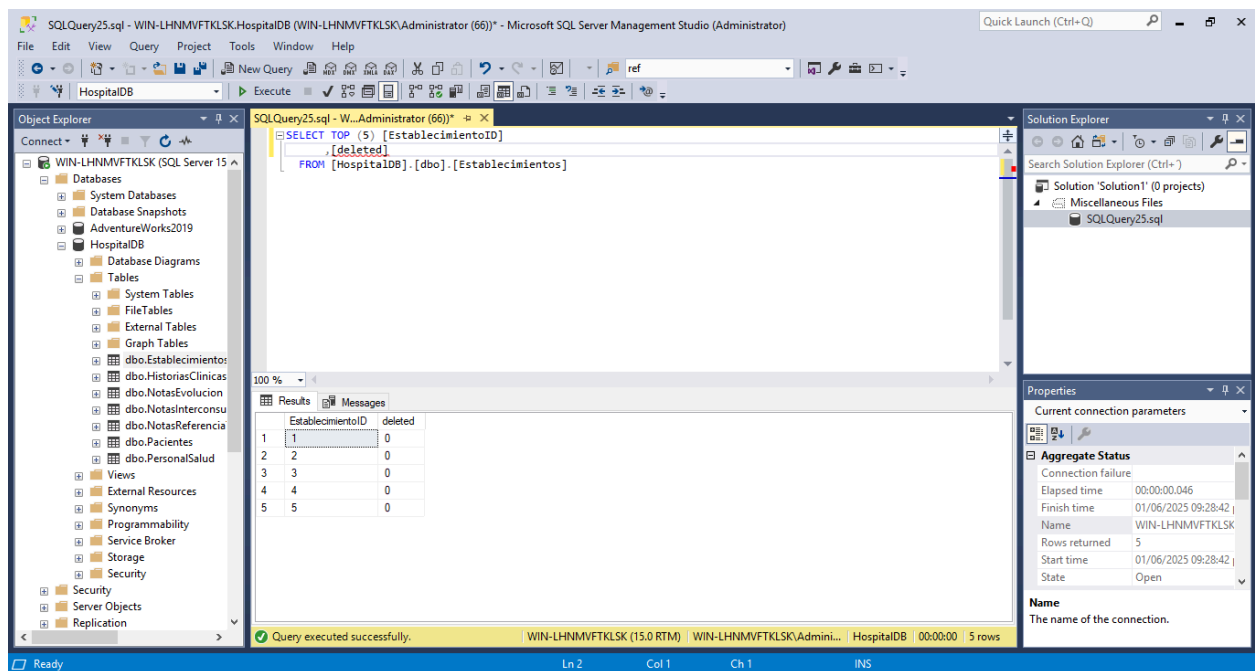
- Resultado de una consulta **SELECT TOP 5 \*, deleted FROM dbo.Pacientes;** mostrando la nueva columna (deleted) con valores 0



- Resultado de una consulta **SELECT TOP 5 \*, deleted FROM dbo.HistoriasClinicas;** mostrando la nueva columna deleted con valores 0



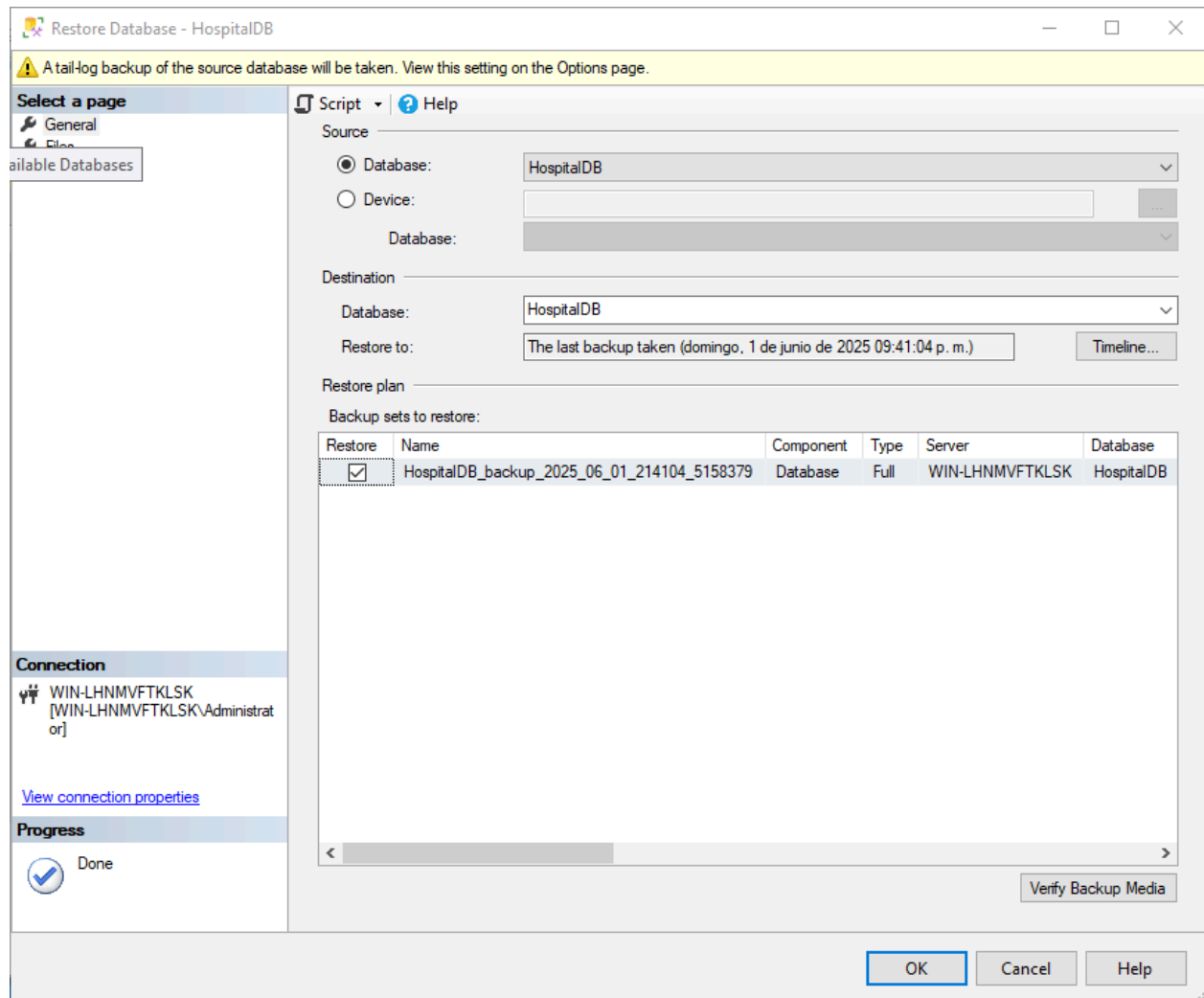
- **Resultado de EXEC sp\_help 'dbo.Establecimientos'; mostrando la definición de la columna deleted en la tabla:**

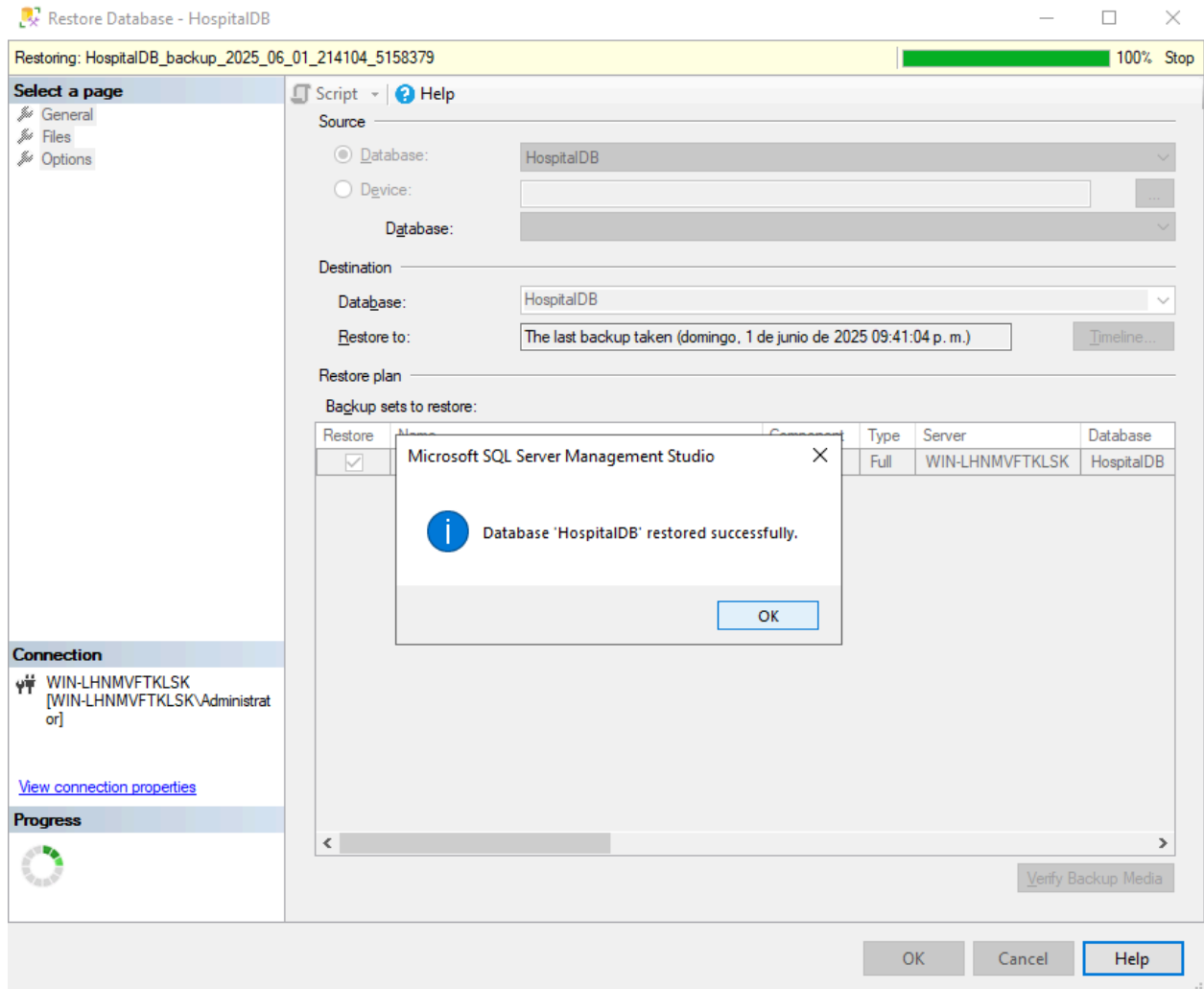


## 6.- Restaurar la base de datos

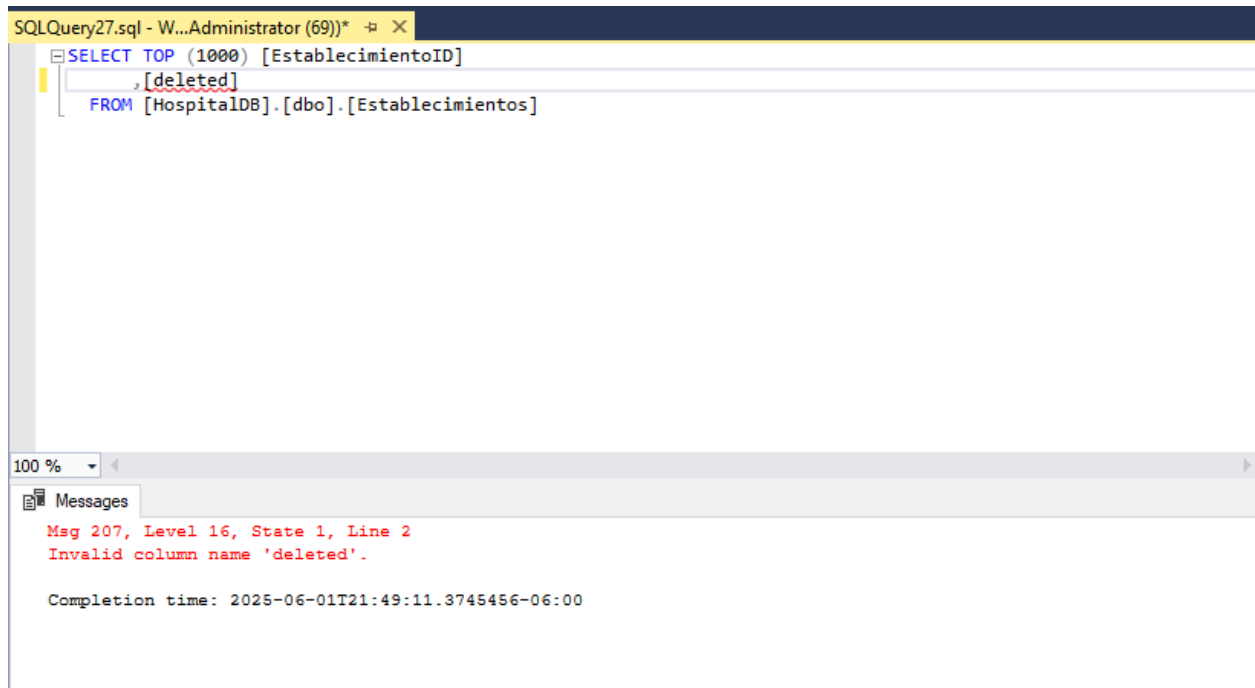
Después de realizar los cambios en la base de datos, se va restaurar la base de datos a su estado previo. Para ello, se va a utilizar un archivo de respaldo completo (.bak) generado, puede ser el hecho durante la asignación 4 (**Plan de Mantenimiento**) o la 3 (**Job de Respaldo**), para esta demostración se usará la hecha por el respaldo generado por la asignación 4. La restauración se realizará mediante el asistente de SSMS.

### Resultados (Capturas):





- Prueba de que se realizó el respaldo:

The screenshot shows the SQL Server Enterprise Manager interface. At the top, a tab is labeled 'SQLQuery27.sql - W...Administrator (69)\*'. The main area contains a T-SQL query: 

```
SELECT TOP (1000) [EstablecimientoID]
, [deleted]
FROM [HospitalDB].[dbo].[Establecimientos]
```

 The word 'deleted' is underlined in red. Below the query, a 'Messages' pane shows an error: 'Msg 207, Level 16, State 1, Line 2 Invalid column name 'deleted''. At the bottom, it says 'Completion time: 2025-06-01T21:49:11.3745456-06:00'.

## Conclusiones

A lo largo de esta asignación, se pusieron en práctica diversas tareas fundamentales en la administración de bases de datos, en específico en un ambiente de SQL Server. Desde el diseño y creación inicial de la base de datos (**HospitalDB**) conforme a la norma y lo postulado por la asignación, pasando por la gestión de la seguridad con usuarios y roles, la automatización de respaldos mediante Jobs y Planes de Mantenimiento, hasta la modificación estructural y la crucial tarea de restauración.

Estas actividades permiten una mejor comprensión sobre el ciclo de vida de una base de datos y la importancia de tener buenas prácticas para asegurar su integridad, disponibilidad y seguridad. El uso de SQL Server Management Studio y T-SQL ha sido fundamental para llevar a cabo cada una de las tareas asignadas.

Algo que me gustaría hacer énfasis es en la importancia de tener una buena configuración, debido a que, durante la realización de esta asignación me tope múltiples problemas debido a una mala configuración del gestor (SSMS) y tuve que buscar todos los errores para poder hacer un poco de *troubleshooting* para poder hacer que todo funcionara.