



## TripAdvisor

Proyecto realizado por Adrián Pardiñas García

### PRESENTACIÓN

Este proyecto está basado en la web estadounidense que proporciona reseñas de contenido relacionado con viajes, como hoteles o restaurantes.

Para ello, es necesario una máquina Windows10 con SQLServer con el SSMS para la administración de la base de datos y SQLDeveloper para su inicial creación, empleando el DataModeler.

Del mismo modo, se creará también una maquina virtual en WindowsServer2016, la cual también tendrá el SQLServer (más el SSMS) y una máquina Linux que tendrá instalado el servidor de SQL, y en este caso el Azure como GUI.

### ÍNDICE

[Índice](#)

[Presentación](#)

[Configuración de maquinas virtuales](#)

[Windows](#)

[Instalación SQL Server](#)

[Instalación SSMS \(SQL Server Managment Studio\)](#)

[Firewall de W10 y Windows Server 2016](#)

[Instalación Oracle Express](#)

[Linux \(Ubuntu\)](#)

[Instalación SQL Server](#)

[Conexiones](#)

[Windows 10 desde CMD a Oracle SQL Developer](#)

[Oracle SQL Developer a SQL Server](#)

[Windows Server2016 a Windows 10](#)

[Azure a Windows 10](#)

[Usuario HR](#)

[Instalar bases de datos de ejemplo](#)

[Pubs desde script](#)

[Northwind con ATTACH](#)

[Adventureworks2017 desde BACKUP](#)

[WideWorldImporters con BACPAC](#)

[Modelado de base de datos](#)

[Modelo Lógico](#)

[Modelo Relacional](#)

[Transformación de SQL Developer a SQL Server](#)

[Modelo Físico](#)

[Importar de SQL Server a SQL Data Modeler](#)

[Administración de bases de datos](#)

[FileTable](#)

[FileStream](#)

[Visualización mediante POWER BI](#)

[Bases de datos contenidas](#)

[Planes de mantenimiento](#)

[Particiones](#)

[Tablas temporales](#)

[Tablas "In Memory"](#)

[GIT](#)

[Creación del repositorio](#)

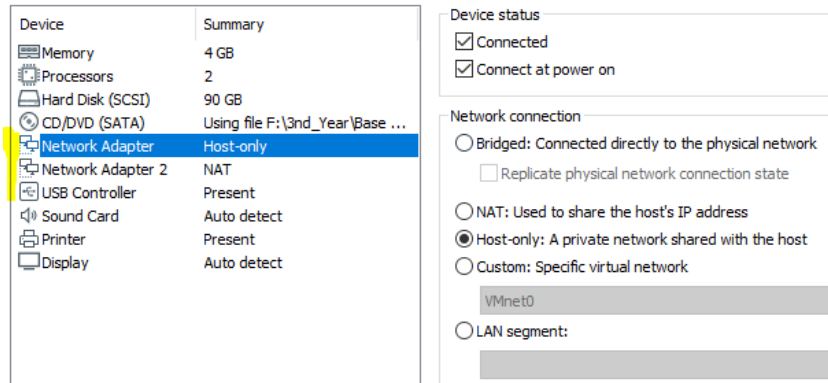
[Enlace a mi repositorio GIT](#)

---

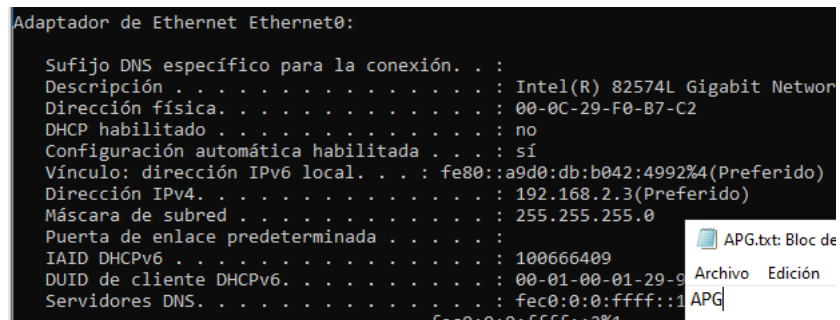
## CONFIGURACIÓN DE MÁQUINAS VIRTUALES

### Windows

Lo primero de todo es configurar de la red en VMWare de todas las máquinas, ya que todas ellas deben de estar en Host-only para que tengan conexión entre sí.



Configuración del adaptador de red en Windows 10



## Instalación de SQL Server

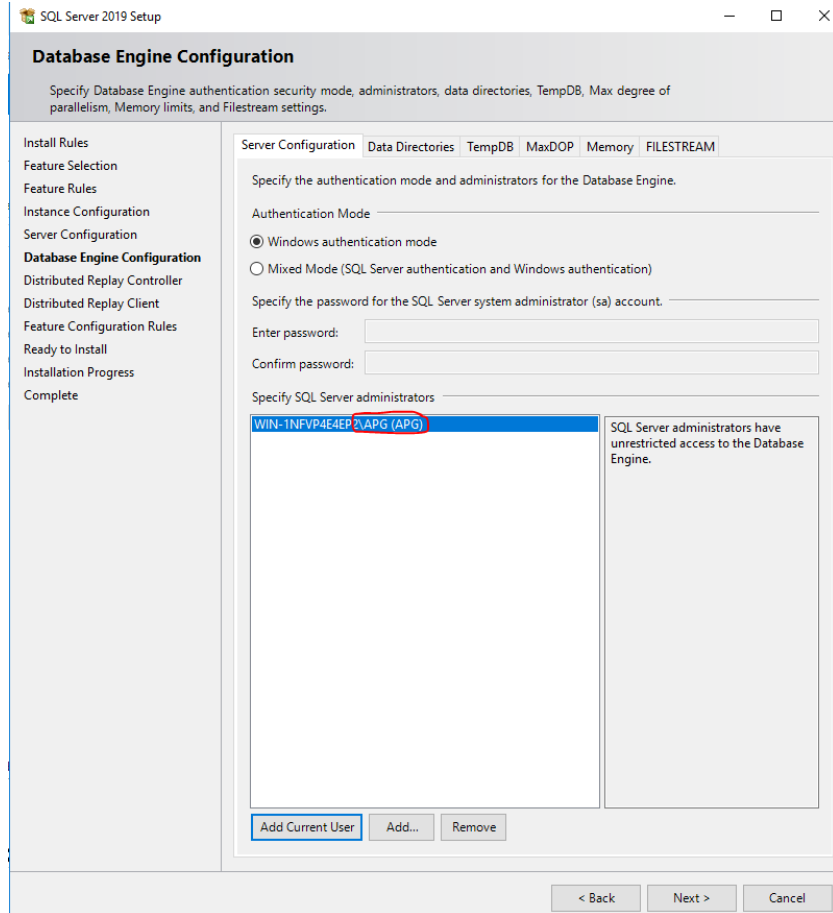
Para la instalación del software se ha seguido los pasos del siguiente enlace

### Cómo instalar y configurar SQL SERVER paso a paso - EXCEleINFO

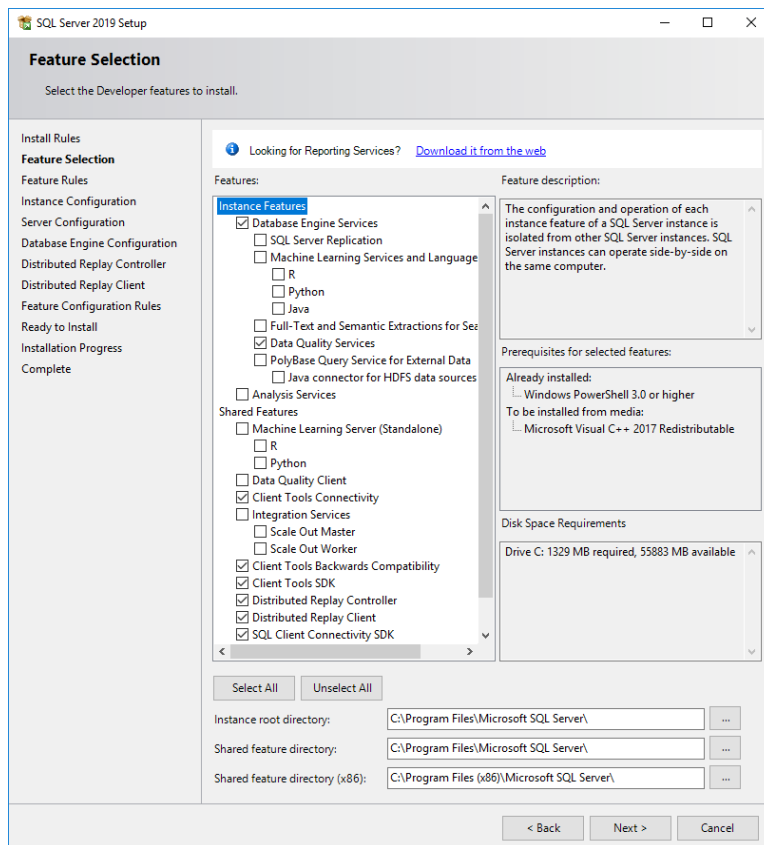
El tema de Excel es una base de datos o no, ya no es trascendencia, porque ya nos quedó claro que no. Excel no es base de datos, pero sí es la herramienta EXCEleINFO Cómo instalar y configurar SQL SERVER paso a paso -

<https://www.exceleinfo.com/como-instalar-y-configurar-sql-server-paso-a-paso/>





En este apartado se han hecho ✓ en todas las casillas de Cliente además de otras importantes



## Instalación del SSMS (SQL Server Management Studio)

Es necesario para trabajar la instalación del SSMS, que es una interfaz GUI para manejar el servidor de SQL.

Aquí figuran unos pasos a seguir para la instalación.

### Cómo instalar y configurar SQL SERVER paso a paso - EXCELeINFO

El tema de Excel es una base de datos o no, ya no es trascendencia, porque ya nos quedó claro que no. Excel no es base de datos, pero sí es la herramienta EXCELeINFO Cómo instalar y configurar SQL SERVER paso a paso -

[https://www.exceleinfo.com/como-instalar-y-configurar-sql-server-paso-a-paso/#:~:text=Instalar%20SQL%20Server%20Management%20Studio%20\(SSMS\)](https://www.exceleinfo.com/como-instalar-y-configurar-sql-server-paso-a-paso/#:~:text=Instalar%20SQL%20Server%20Management%20Studio%20(SSMS))



## Configuración del Firewall de W10 y Windows Server

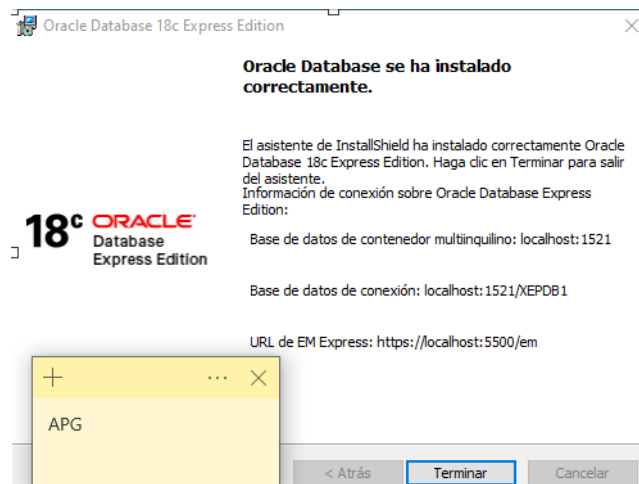
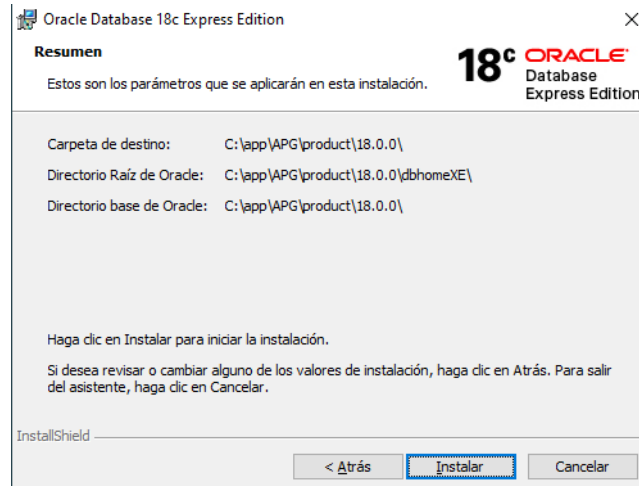
Para permitir una perfecta conexión entre distintas máquinas al servidor de SQL Server, es conveniente habilitar una regla de entrada y otra de salida para el SQL\_Server en la configuración de Windows Firewall, abriendo el puerto 1433. Otra opción, no recomendada, es deshabilitar el Firewall.

Reglas de entrada												
Nombre	Grupo	Perfil	Habilitado	Acción	Invaldar	Programa	Dirección local	Dirección remota	Protocolo	Puerto local	Puerto remoto	
Java(TM) Platform SE binary		Privado	Sí	Permitir	No	C:\app\app\product\18.0.0\dbhome...	Cualquiera	Cualquiera	TCP	Cualquiera	Cualquiera	
Java(TM) Platform SE binary		Privado	Sí	Permitir	No	C:\app\app\product\18.0.0\dbhome...	Cualquiera	Cualquiera	UDP	Cualquiera	Cualquiera	
SQL_Server		Todo	Sí	Permitir	No	Cualquiera	Cualquiera	Cualquiera	TCP	1433	Cualquiera	
@FirewallAPI.dll - 80201	@FirewallAPI.dll - 80200	Todo	Sí	Permitir	No	%SystemRoot%\system32\svchost.e...	Cualquiera	Subred local	TCP	554, 8554-85...	Cualquiera	

Reglas de salida										
Nombre	Grupo	Perfil	Habilitado	Acción	Invalidar	Programa	Dirección local	Dirección remota	Protocolo	Puerto local
SQL_Server		Todo	Sí	Permitir	No	Cualquiera	Cualquiera	Cualquiera	TCP	Cualquiera
@FirewallAPI.dll - 80204	@FirewallAPI.dll - 80200	Todo	Sí	Permitir	No	%SystemRoot%\system32\svch...	Cualquiera	Subred local	TCP	Cualquiera

## Instalación de Oracle Express

Es prácticamente un proceso de “next next”



## LINUX: Ubuntu

Del mismo modo que se ha hecho con Windows, ahora se configura la red de Ubuntu, cuya IP es 192.168.2.4 (dentro de la misma red que las de Windows).

```
apg@apg-virtual-machine: ~/Desktop
3: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state U
up default qlen 1000
    link/ether 00:0c:29:d8:32:aa brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.2.4/24 brd 192.168.2.255 scope global noprefixroute ens38
        valid_lft forever preferred_lft forever
    inet6 fe80::d6b2:8559:737f:2cfe/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
apg@apg-virtual-machine:~/Desktop$
```

## Instalación de SQL Server en Ubuntu

Se comprueba que funciona el servicio del servidor SQL Server, que previamente se instaló continuando un tutorial.

### Ubuntu: Instalación de SQL Server en Linux - SQL Server

Se aplica a: SQL Server (todas las versiones admitidas): Linux En este inicio rápido, instalará SQL Server 2017 en Ubuntu 16.04/18.04. Después, se conectará con sqlcmd para crear la primera base de datos y ejecutar consultas. En este inicio rápido, instalará SQL Server 2019 en Ubuntu 16.04, 18.04 o 20.04.

 <https://docs.microsoft.com/es-es/sql/linux/quickstart-install-connect-ubuntu?view=sql-server-ver15>

```
apg@apg-virtual-machine: ~
apg@apg-virtual-machine:~$ systemctl status mssql-server
● mssql-server.service - Microsoft SQL Server Database Engine
   Loaded: loaded (/lib/systemd/system/mssql-server.service; enabled; vendor
   Active: active (running) since Tue 2022-02-22 13:42:53 CET; 14min ago
     Docs: https://docs.microsoft.com/en-us/sql/linux
   Main PID: 844 (sqlservr)
      Tasks: 132
     Memory: 445.3M
    CGroup: /system.slice/mssql-server.service
            └─ 844 /opt/mssql/bin/sqlservr
               1637 /opt/mssql/bin/sqlservr

feb 22 13:43:18 apg-virtual-machine sqlservr[1637]: [66B blob data]
feb 22 13:43:19 apg-virtual-machine sqlservr[1637]: [75B blob data]
feb 22 13:43:19 apg-virtual-machine sqlservr[1637]: [96B blob data]
feb 22 13:43:19 apg-virtual-machine sqlservr[1637]: [100B blob data]
feb 22 13:43:19 apg-virtual-machine sqlservr[1637]: [71B blob data]
feb 22 13:43:19 apg-virtual-machine sqlservr[1637]: [124B blob data]
feb 22 13:44:14 apg-virtual-machine sqlservr[1637]: [73B blob data]
feb 22 13:44:14 apg-virtual-machine sqlservr[1637]: [145B blob data]
feb 22 13:48:37 apg-virtual-machine sqlservr[1637]: [156B blob data]
feb 22 13:48:37 apg-virtual-machine sqlservr[1637]: [195B blob data]
lines 1-21/21 (END)
```

Nos logueamos en el terminal dentro de SQL y podemos comprobar como vemos las bases de datos, donde habíamos añadido "prueba1" a mayores de las bases de datos de sistema que vienen por defecto.

```

apg@apg-virtual-machine:~$ sqlcmd -S localhost -U SA -P 'Abcd1234.'
1> select name from sys.databases
2> go
name
-----
master
tempdb
model
msdb
prueba1

(5 rows affected)
1>

```

## CONEXIONES

### Conexión desde CMD a Oracle Database

```

C:\Users\APG>sqlplus

Microsoft Windows [Versión 10.0.19043.1288]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\APG>sqlplus

SQL*Plus: Release 18.0.0.0.0 - Production on Jue Nov 4 19:12:45 2021
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Introduzca el nombre de usuario: SYS / AS SYSDBA
Introduzca la contraseña:

Conectado a:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

SQL> show user
USER es "SYS"
SQL>

```

### Oracle SQL Developer a SQL Server

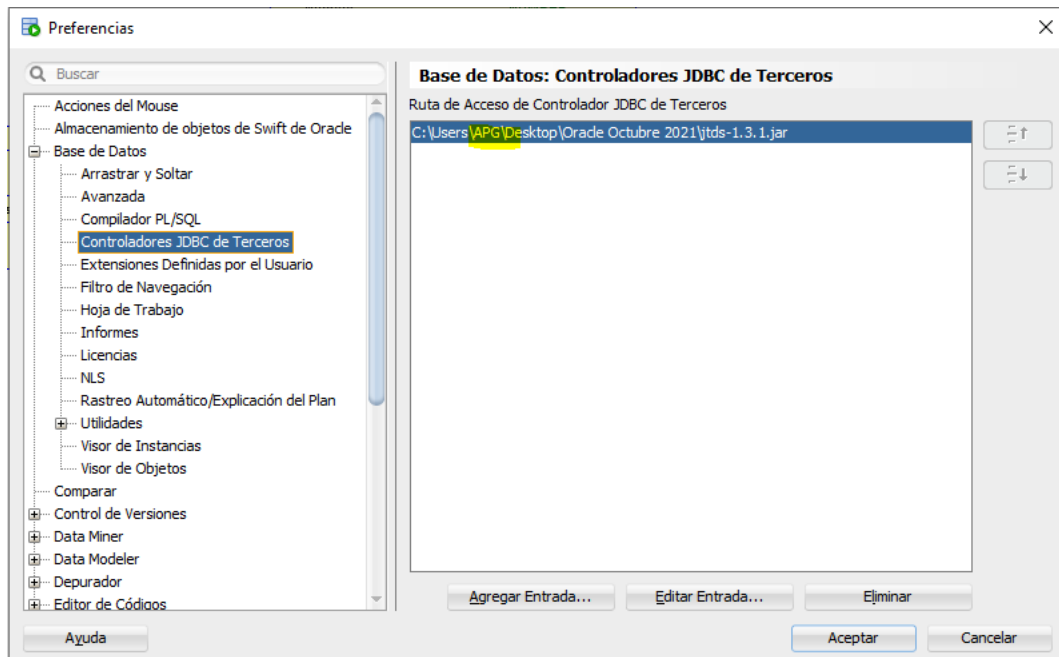
Primero de todo es necesario introducir los drivers para que en Tipo de Base de Datos salga la opción de SQLServer. Se sigue este enlace como guía.

#### Third Party Databases Drivers in Oracle SQL Developer

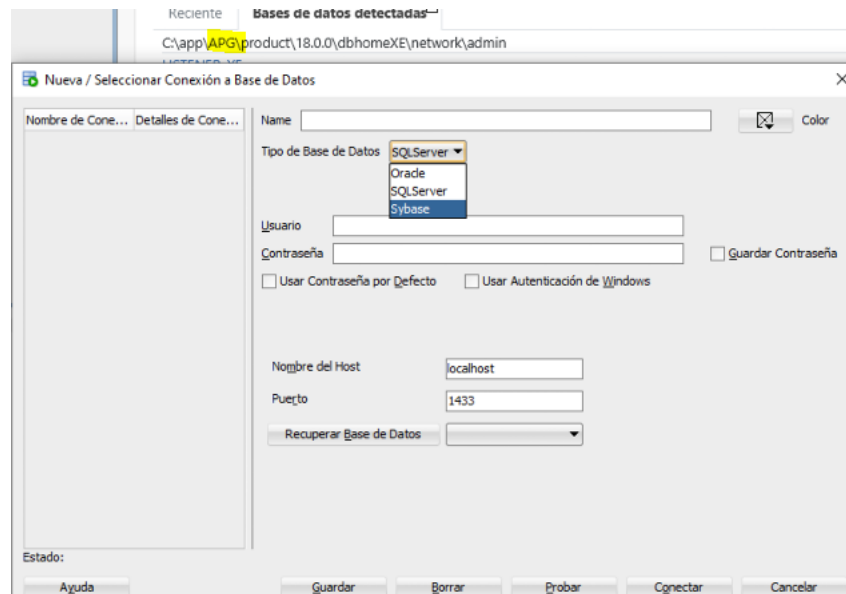
Oracle SQL Developer 1.5 (1.5.0.53.38) supports browsing a selection of third party databases; Microsoft Access, Microsoft SQL Server, MySQL and Sybase. Before accessing these databases, you need to download and load the correct third party drivers. You can do this manually or using Check for Updates.

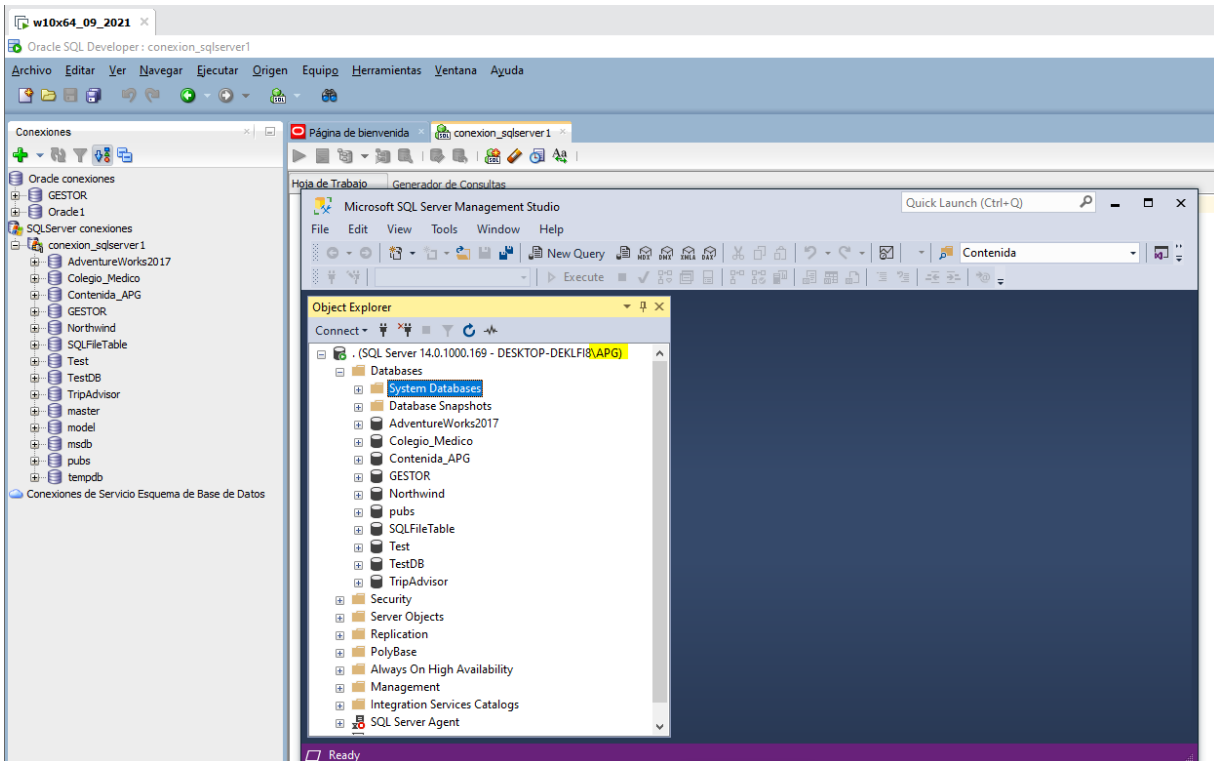
<https://www.oracle.com/database/technologies/appdev/sqldev/thirdparty.html>





Después ya aparece dicha función, y entonces se puede realizar la conexión





## Conexión Windows Server 2016 a SQL Server de Windows 10

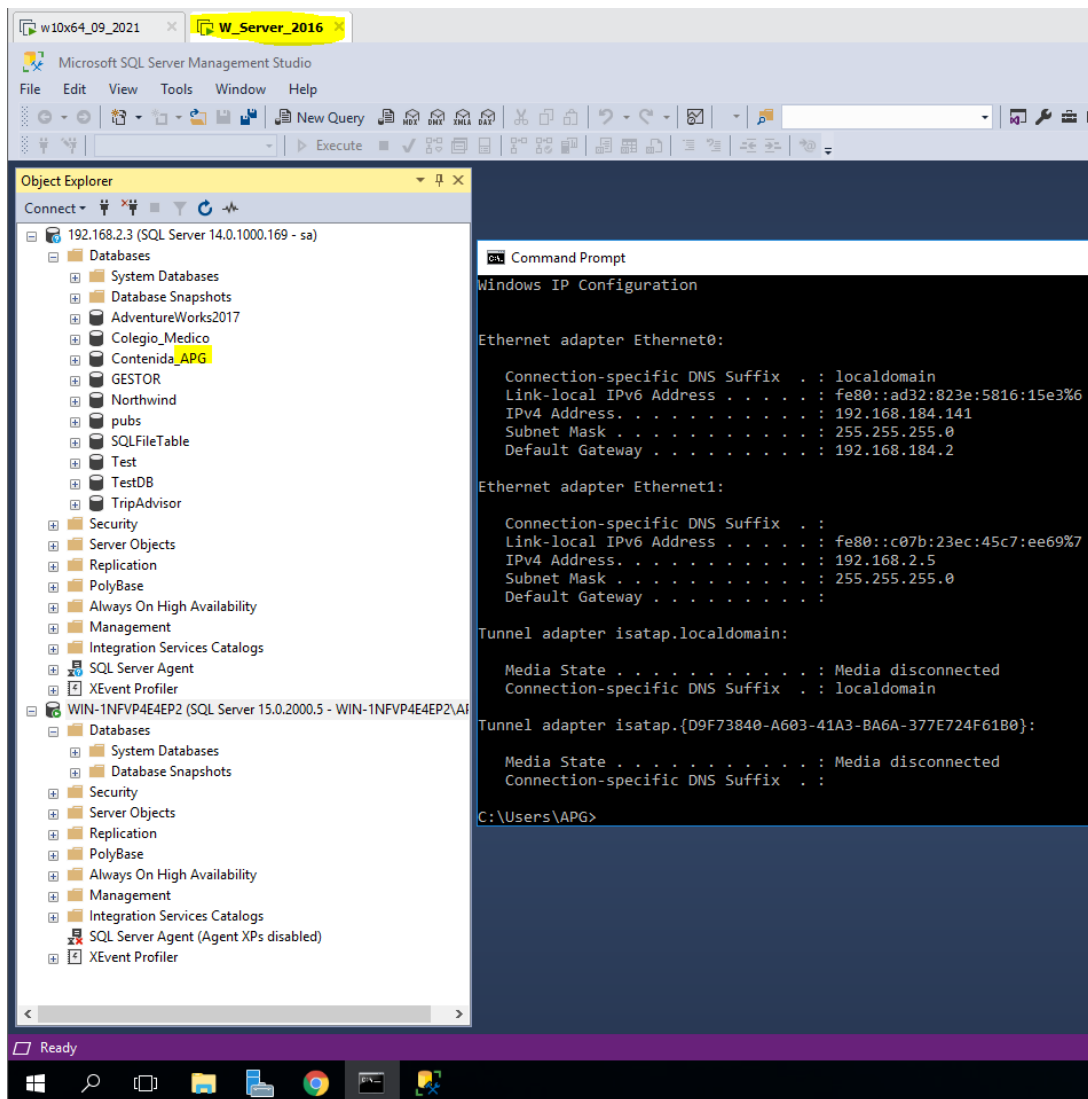
Desde Windows Server podemos ver como nos podemos conectar a la maquina de windows server, cuya IP era 192.168.2.3 que se reflejó en el apartado anterior.

```

Adaptador de Ethernet Ethernet0:

Sufijo DNS específico para la conexión. . . :
Descripción . . . . . : Intel(R) 82574L Gigabit Network
Dirección física. . . . . : 00-0C-29-F0-B7-C2
DHCP habilitado . . . . . : no
Configuración automática habilitada . . . : sí
Vínculo: dirección IPv6 local. . . : fe80::a9d0:db:b042:4992%4(Preferido)
Dirección IPv4. . . . . : 192.168.2.3(Preferido)
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . :
IAID DHCPv6 . . . . . : 100666409
DUID de cliente DHCPv6. . . . . : 00-01-00-01-29-9
Servidores DNS. . . . . : fec0:0:0:ffff::1

```



## Conexión Azure a Windows 10

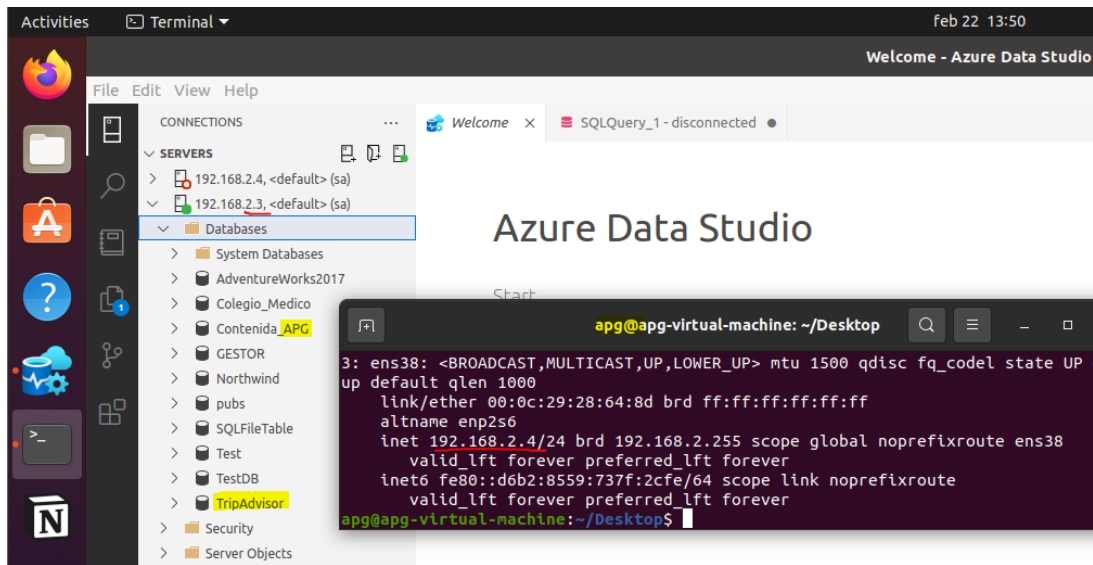
Instalamos Azure Data Studio a través de la siguiente guía:

### Descarga e instalación de Azure Data Studio - Azure Data Studio

Azure Data Studio es una herramienta de base de datos multiplataforma para profesionales de datos que usan plataformas de datos locales y en la nube en Windows, macOS y Linux. Azure Data Studio ofrece una experiencia de editor moderna con IntelliSense, fragmentos de código, integración del control de código fuente y <https://docs.microsoft.com/es-es/sql/azure-data-studio/download-azure-data-studio?view=sql-server-ver15#linux-installation>



Continuamos dentro de Ubuntu y ahora a través de Azure Data Studio nos conectamos a la máquina virtual de Windows 10 (192.168.2.3) y podemos ver como aparecen las bases de datos de la otra máquina, donde ya había creado previamente la del propio proyecto, TripAdvisor.



## Usuario HR

Al instalar Oracle Express, el usuario HR viene por defecto deshabilitado y hay que "activarlo mediante los siguientes pasos

```

SQL> alter user HR identified by HR account unlock;

User altered.

SQL> select username, account_status from dba_users;

```

USERNAME	ACCOUNT_STATUS
SYS	OPEN
SYSTEM	OPEN
XS\$NULL	EXPIRED & LOCKED
LBACSYS	LOCKED
OUTLN	EXPIRED & LOCKED
DBSNMP	EXPIRED & LOCKED
APPOSSYS	EXPIRED & LOCKED
DBSFUSER	EXPIRED & LOCKED
GGSYS	EXPIRED & LOCKED
ANONYMOUS	EXPIRED & LOCKED
HR	OPEN
...	...

De esta forma ya logramos loguearnos

Nueva / Seleccionar Conexión a Base de Datos

Nombre de Cone...	Detalles de Cone...
conexion_sqlserv...	@//localhost:143...
GESTOR	GESTOR@//local...
Oracle1	system@//localh...

Nombre:

Tipo de Base de Datos:

**Información de usuario**

Tipo de autenticación:

Usuario:

Contraseña:

Tipo de Conexión:

**Detalles**

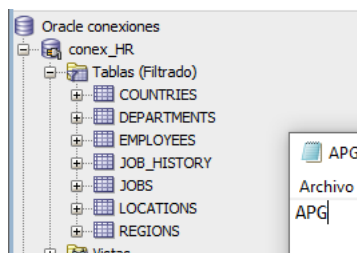
Nombre del Host:

Puerto:

☐ SID

☒ Nombre del Servicio

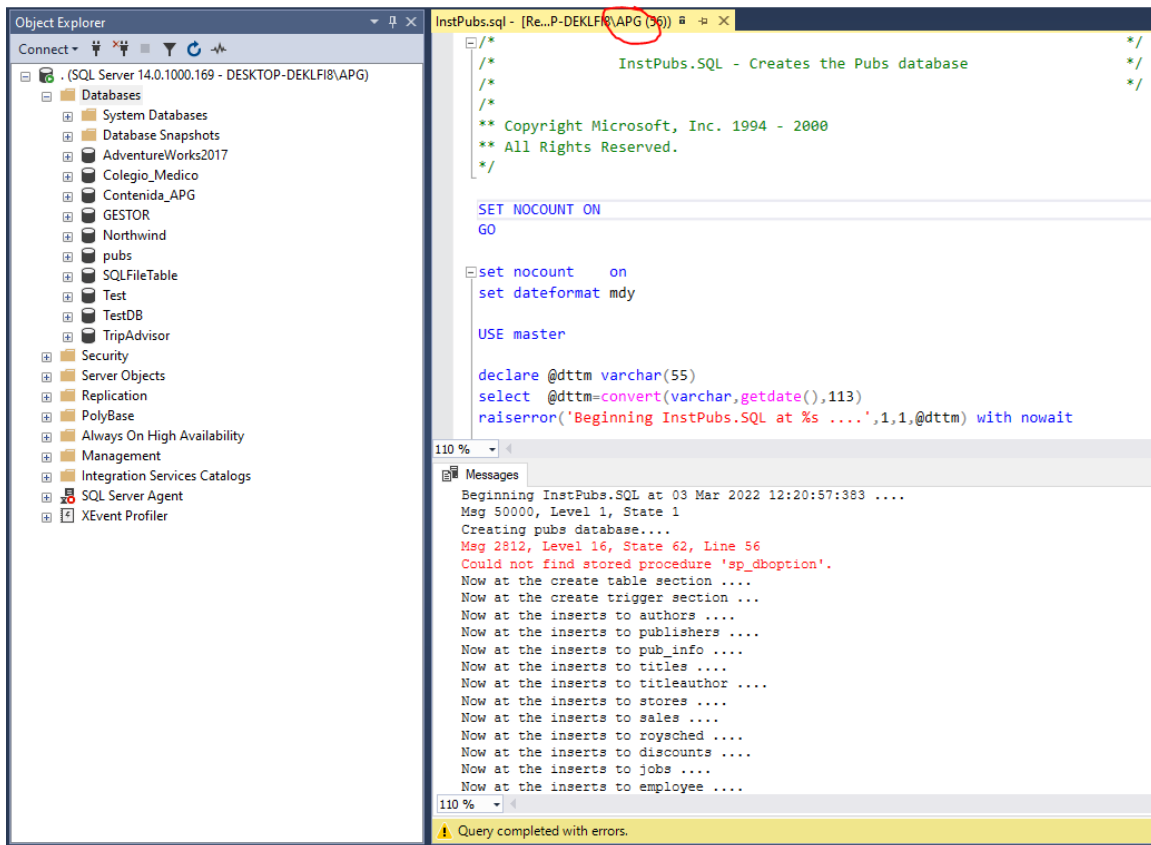
Estado: Correcto



## INSTALAR BASES DE DATOS DE EJEMPLO

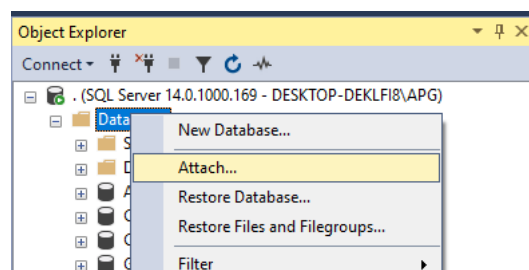
### Pubs desde Script

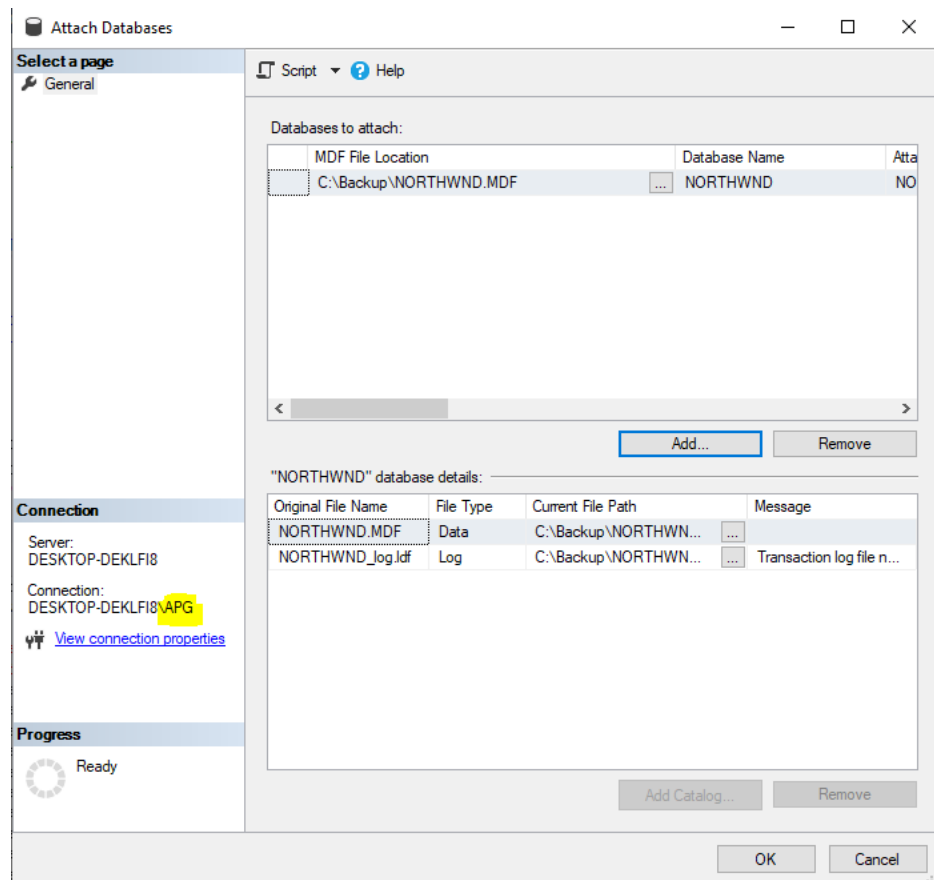
Ejecutar todo el script una vez abierto. Genera unos errores pero lo abre sin problema como se puede ver en el explorador de objetos.



## Northwind con Attach

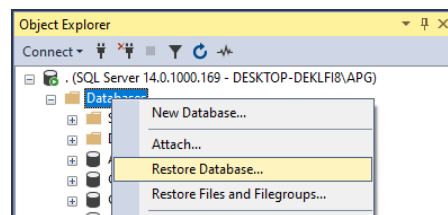
Botón derecho sobre database y elegir "Attach"

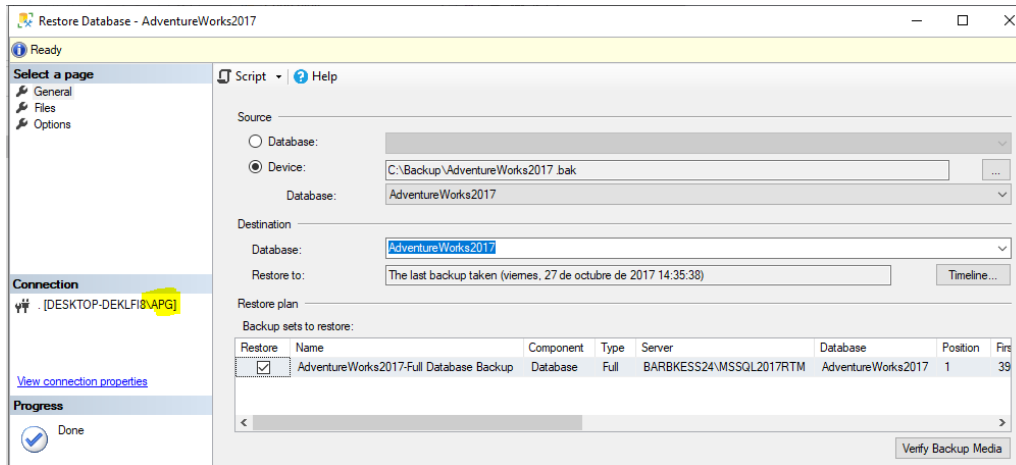




## AdventureWorks2017 desde Backup

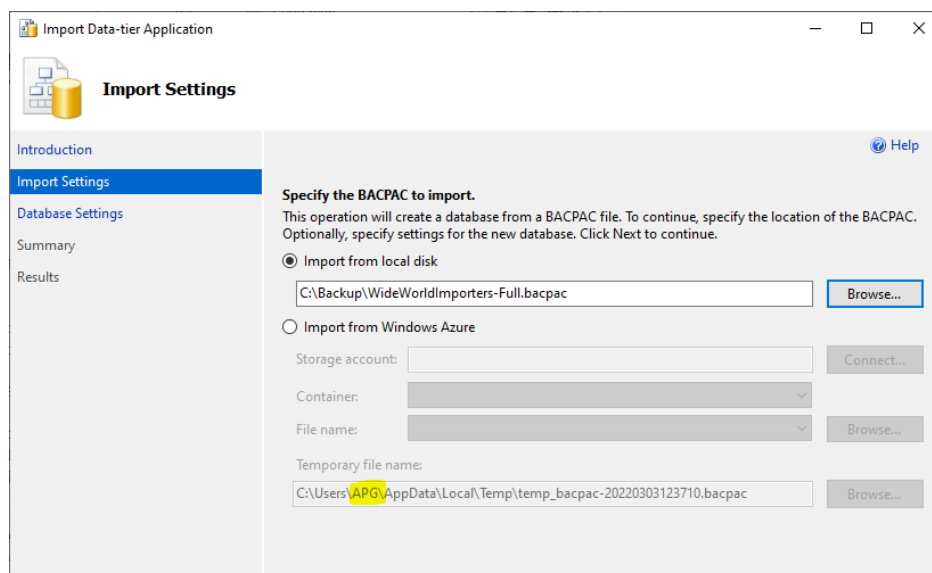
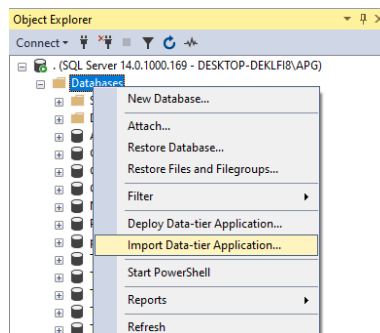
Como en el proceso anterior, botón derecho sobre database y en este caso "Restore Database". Se restablecerá desde un archivo .bak



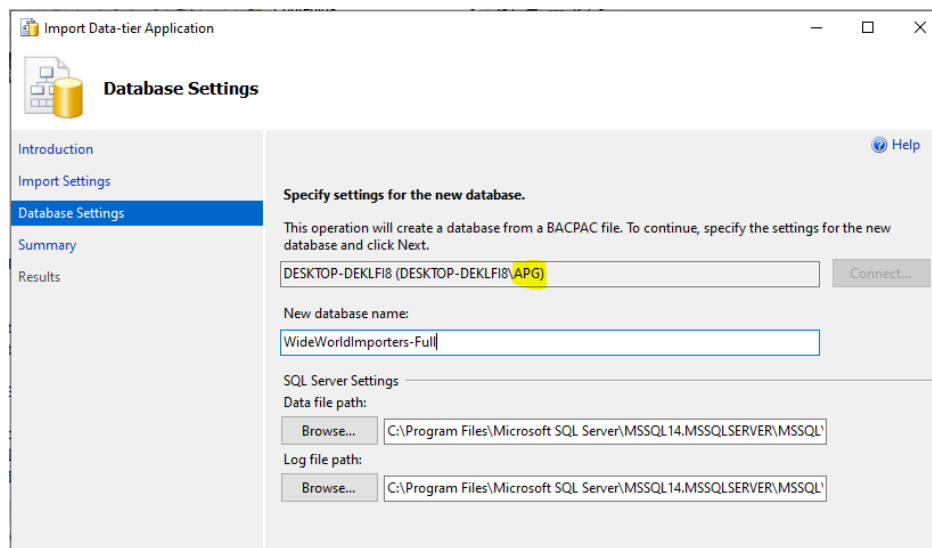


## WideWorldImporters desde BACPAC

Sencillos pasos como en los procesos anteriores. Archivo .bacpac



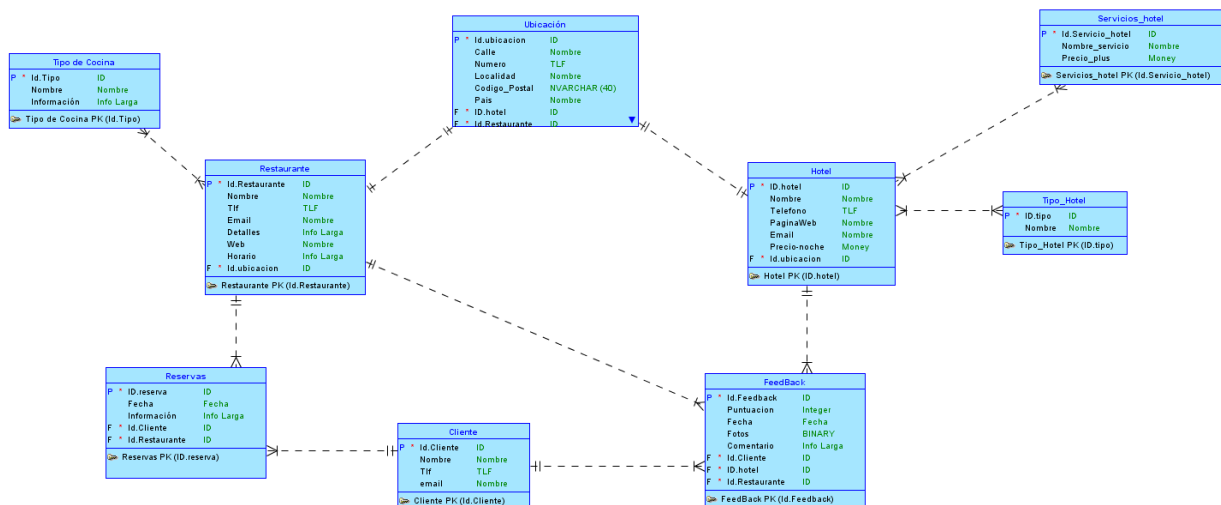




## MODELADO DE DATOS

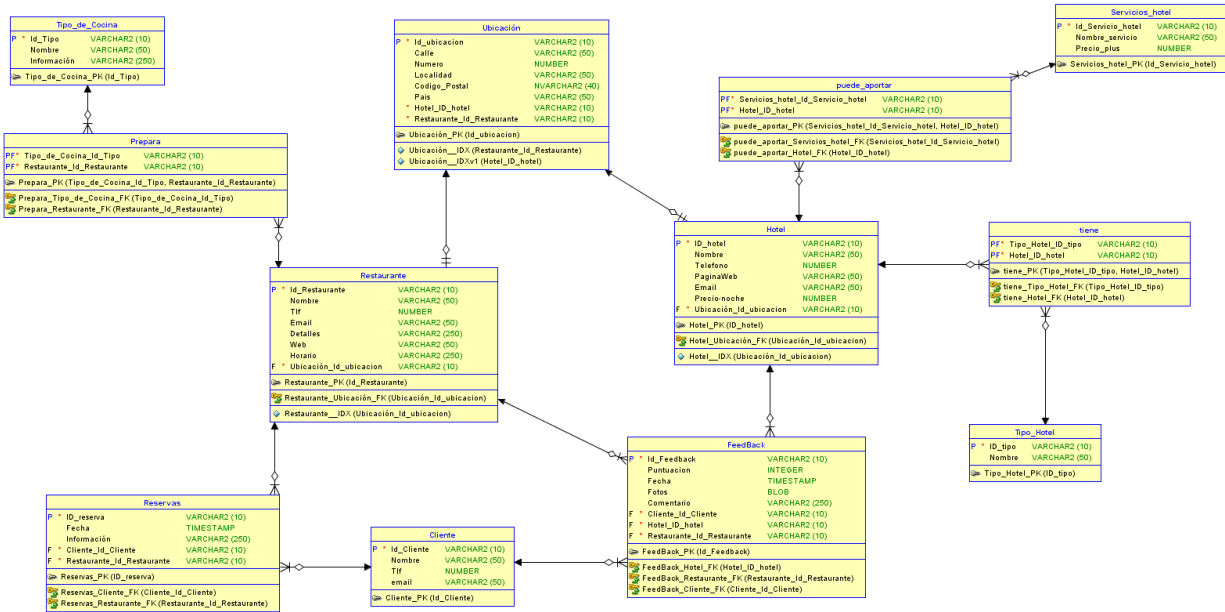
### Modelo Lógico

La base de datos consta de 9 tablas.



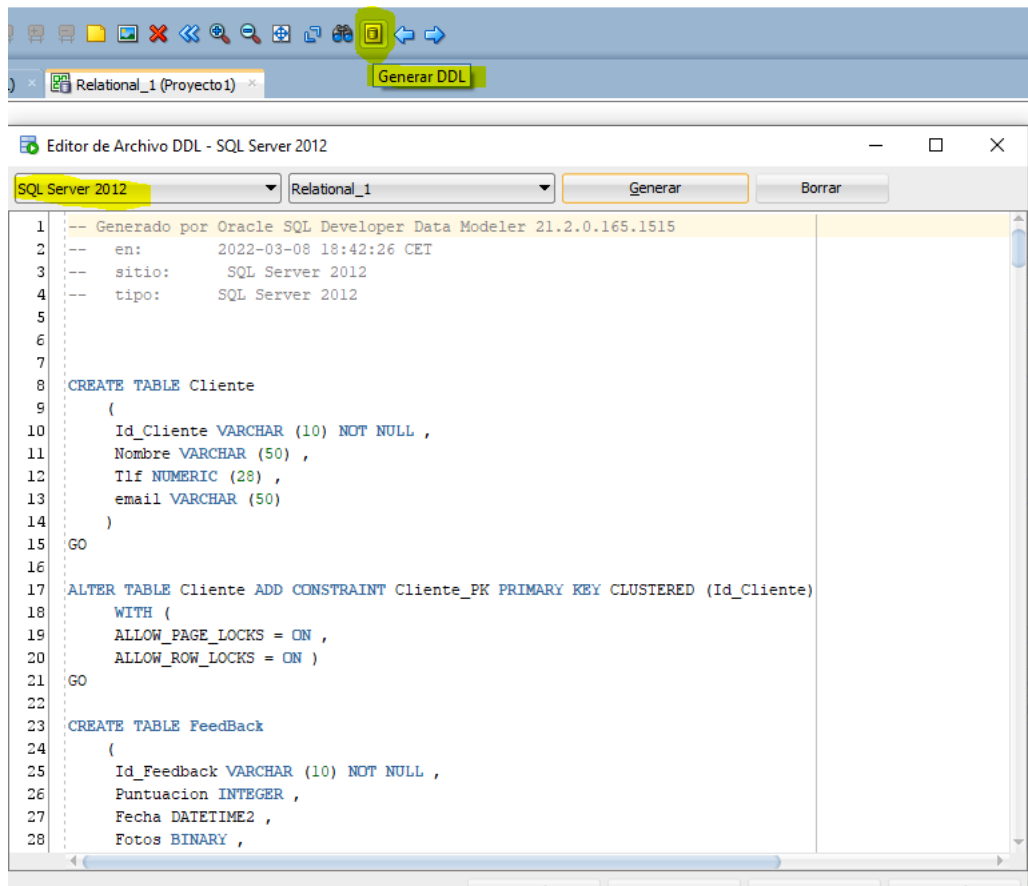
## Modelo Relacional

Al hacer el paso al modelo relacional, al haber 3 relaciones de tipo N:M , genera nuevas tablas.



## De SQL Developer a SQL Server

A través del Data Modeler podemos generar el script en formato SQL Server 2012.



## Script base de datos SQL Server

Una vez hecho, se seleccionaría todo el texto y se copiaría en un script .sql el cual ya se puede ejecutar. Para ello, primeramente, hay que crear la base de datos, y ponerse en uso de ella, porque sino daría un error.

```

-- Generado por Oracle SQL Developer Data Modeler 21.2.0.165.1515
-- en:      2022-02-10 20:17:15 CET
-- sitio:   SQL Server 2012
-- tipo:    SQL Server 2012
--drop database if exists TripAdvisor
--create database TripAdvisor

--use TripAdvisor

CREATE TABLE Cliente
(
    Id_Cliente VARCHAR (10) NOT NULL ,
    Nombre VARCHAR (50) ,
    Tlf NUMERIC (28) ,
    email VARCHAR (50)
)
GO

ALTER TABLE Cliente ADD CONSTRAINT Cliente_PK PRIMARY KEY CLUSTERED (Id_Cliente)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

```

```

CREATE TABLE FeedBack
(
    Id_Feedback VARCHAR (10) NOT NULL ,
    Puntuacion INTEGER ,
    Fecha DATETIME2 ,
    Fotos BINARY ,
    Comentario VARCHAR (250) ,
    Cliente_Id_Cliente VARCHAR (10) NOT NULL ,
    Hotel_ID_hotel VARCHAR (10) NOT NULL ,
    Restaurante_Id_Restaurante VARCHAR (10) NOT NULL
)
GO

ALTER TABLE FeedBack ADD CONSTRAINT FeedBack_PK PRIMARY KEY CLUSTERED (Id_Feedback)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Hotel
(
    ID_hotel VARCHAR (10) NOT NULL ,
    Nombre VARCHAR (50) ,
    Telefono NUMERIC (28) ,
    PaginaWeb VARCHAR (50) ,
    Email VARCHAR (50) ,
    "Precio-noche" MONEY ,
    Ubicación_Id_ubicacion VARCHAR (10) NOT NULL
)
GO

CREATE UNIQUE NONCLUSTERED INDEX
    Hotel__IDX ON Hotel
(
    Ubicación_Id_ubicacion
)
GO

ALTER TABLE Hotel ADD CONSTRAINT Hotel_PK PRIMARY KEY CLUSTERED (ID_hotel)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Prepara
(
    Tipo_de_Cocina_Id_Tipo VARCHAR (10) NOT NULL ,
    Restaurante_Id_Restaurante VARCHAR (10) NOT NULL
)
GO

ALTER TABLE Prepara ADD CONSTRAINT Prepara_PK PRIMARY KEY CLUSTERED (Tipo_de_Cocina_Id_Tipo, Restaurante_Id_Restaurante)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE puede_aportar
(
    Servicios_hotel_Id_Servicio_hotel VARCHAR (10) NOT NULL ,
    Hotel_ID_hotel VARCHAR (10) NOT NULL
)
GO

ALTER TABLE puede_aportar ADD CONSTRAINT puede_aportar_PK PRIMARY KEY CLUSTERED (Servicios_hotel_Id_Servicio_hotel, Hotel_ID_hotel)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Reservas
(
    ID_reserva VARCHAR (10) NOT NULL ,
    Fecha DATETIME2 ,
    Información VARCHAR (250) ,
    Cliente_Id_Cliente VARCHAR (10) NOT NULL ,
    Restaurante_Id_Restaurante VARCHAR (10) NOT NULL
)
GO

```

```

ALTER TABLE Reservas ADD CONSTRAINT Reservas_PK PRIMARY KEY CLUSTERED (ID_reserva)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Restaurante
(
    Id_Restaurante VARCHAR (10) NOT NULL ,
    Nombre VARCHAR (50) ,
    Tlf NUMERIC (28) ,
    Email VARCHAR (50) ,
    Detalles VARCHAR (250) ,
    Web VARCHAR (50) ,
    Horario VARCHAR (250) ,
    Ubicación_Id_ubicacion VARCHAR (10) NOT NULL
)
GO

CREATE UNIQUE NONCLUSTERED INDEX
    Restaurante__IDX ON Restaurante
(
    Ubicación_Id_ubicacion
)
GO

ALTER TABLE Restaurante ADD CONSTRAINT Restaurante_PK PRIMARY KEY CLUSTERED (Id_Restaurante)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Servicios_hotel
(
    Id_Servicio_hotel VARCHAR (10) NOT NULL ,
    Nombre_servicio VARCHAR (50) ,
    Precio_plus MONEY
)
GO

ALTER TABLE Servicios_hotel ADD CONSTRAINT Servicios_hotel_PK PRIMARY KEY CLUSTERED (Id_Servicio_hotel)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE tiene
(
    Tipo_Hotel_ID_tipo VARCHAR (10) NOT NULL ,
    Hotel_ID_hotel VARCHAR (10) NOT NULL
)
GO

ALTER TABLE tiene ADD CONSTRAINT tiene_PK PRIMARY KEY CLUSTERED (Tipo_Hotel_ID_tipo, Hotel_ID_hotel)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Tipo_de_Cocina
(
    Id_Tipo VARCHAR (10) NOT NULL ,
    Nombre VARCHAR (50) ,
    Información VARCHAR (250)
)
GO

ALTER TABLE Tipo_de_Cocina ADD CONSTRAINT Tipo_de_Cocina_PK PRIMARY KEY CLUSTERED (Id_Tipo)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Tipo_Hotel
(
    ID_tipo VARCHAR (10) NOT NULL ,
    Nombre VARCHAR (50)
)

```

```

    )
GO

ALTER TABLE Tipo_Hotel ADD CONSTRAINT Tipo_Hotel_PK PRIMARY KEY CLUSTERED (ID_tipo)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

CREATE TABLE Ubicación
(
    Id_ubicacion VARCHAR (10) NOT NULL ,
    Calle VARCHAR (50) ,
    Numero NUMERIC (28) ,
    Localidad VARCHAR (50) ,
   Codigo_Postal NVARCHAR (40) ,
    Pais VARCHAR (50) ,
    Hotel_ID_hotel VARCHAR (10) NOT NULL ,
    Restaurante_Id_Restaurante VARCHAR (10) NOT NULL
)
GO

CREATE UNIQUE NONCLUSTERED INDEX
    Ubicación__IDX ON Ubicación
(
    Restaurante_Id_Restaurante
)
GO

CREATE UNIQUE NONCLUSTERED INDEX
    Ubicación__IDXv1 ON Ubicación
(
    Hotel_ID_hotel
)
GO

ALTER TABLE Ubicación ADD CONSTRAINT Ubicación_PK PRIMARY KEY CLUSTERED (Id_ubicacion)
WITH (
    ALLOW_PAGE_LOCKS = ON ,
    ALLOW_ROW_LOCKS = ON )
GO

ALTER TABLE FeedBack
    ADD CONSTRAINT FeedBack_Cliente_FK FOREIGN KEY
(
    Cliente_Id_Cliente
)
REFERENCES Cliente
(
    Id_Cliente
)
ON DELETE NO ACTION
ON UPDATE NO ACTION
GO

ALTER TABLE FeedBack
    ADD CONSTRAINT FeedBack_Hotel_FK FOREIGN KEY
(
    Hotel_ID_hotel
)
REFERENCES Hotel
(
    ID_hotel
)
ON DELETE NO ACTION
ON UPDATE NO ACTION
GO

ALTER TABLE FeedBack
    ADD CONSTRAINT FeedBack_Restaurante_FK FOREIGN KEY
( Restaurante_Id_Restaurante)
REFERENCES Restaurante
( Id_Restaurante)
ON DELETE NO ACTION
ON UPDATE NO ACTION
GO

```

```

ALTER TABLE Hotel
    ADD CONSTRAINT Hotel_Ubicación_FK FOREIGN KEY
    (
        Ubicación_Id_ubicacion
    )
    REFERENCES Ubicación
    (
        Id_ubicacion
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE Prepara
    ADD CONSTRAINT Prepara_Restaurante_FK FOREIGN KEY
    (
        Restaurante_Id_Restaurante
    )
    REFERENCES Restaurante
    (
        Id_Restaurante
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE Prepara
    ADD CONSTRAINT Prepara_Tipo_de_Cocina_FK FOREIGN KEY
    (
        Tipo_de_Cocina_Id_Tipo
    )
    REFERENCES Tipo_de_Cocina
    (
        Id_Tipo
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE puede_aportar
    ADD CONSTRAINT puede_aportar_Hotel_FK FOREIGN KEY
    (
        Hotel_ID_hotel
    )
    REFERENCES Hotel
    (
        ID_hotel
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE puede_aportar
    ADD CONSTRAINT puede_aportar_Servicios_hotel_FK FOREIGN KEY
    (
        Servicios_hotel_Id_Servicio_hotel
    )
    REFERENCES Servicios_hotel
    (
        Id_Servicio_hotel
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE Reservas
    ADD CONSTRAINT Reservas_Cliente_FK FOREIGN KEY
    (
        Cliente_Id_Cliente
    )
    REFERENCES Cliente
    (
        Id_Cliente
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE Reservas
    ADD CONSTRAINT Reservas_Restaurante_FK FOREIGN KEY
    (

```

```

        Restaurante_Id_Restaurante
    )
    REFERENCES Restaurante
    (
        Id_Restaurante
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE Restaurante
    ADD CONSTRAINT Restaurante_Ubicación_FK FOREIGN KEY
    (
        Ubicación_Id_ubicacion
    )
    REFERENCES Ubicación
    (
        Id_ubicacion
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

ALTER TABLE tiene
    ADD CONSTRAINT tiene_Hotel_FK FOREIGN KEY
    (
        Hotel_ID_hotel
    )
    REFERENCES Hotel
    (
        ID_hotel
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

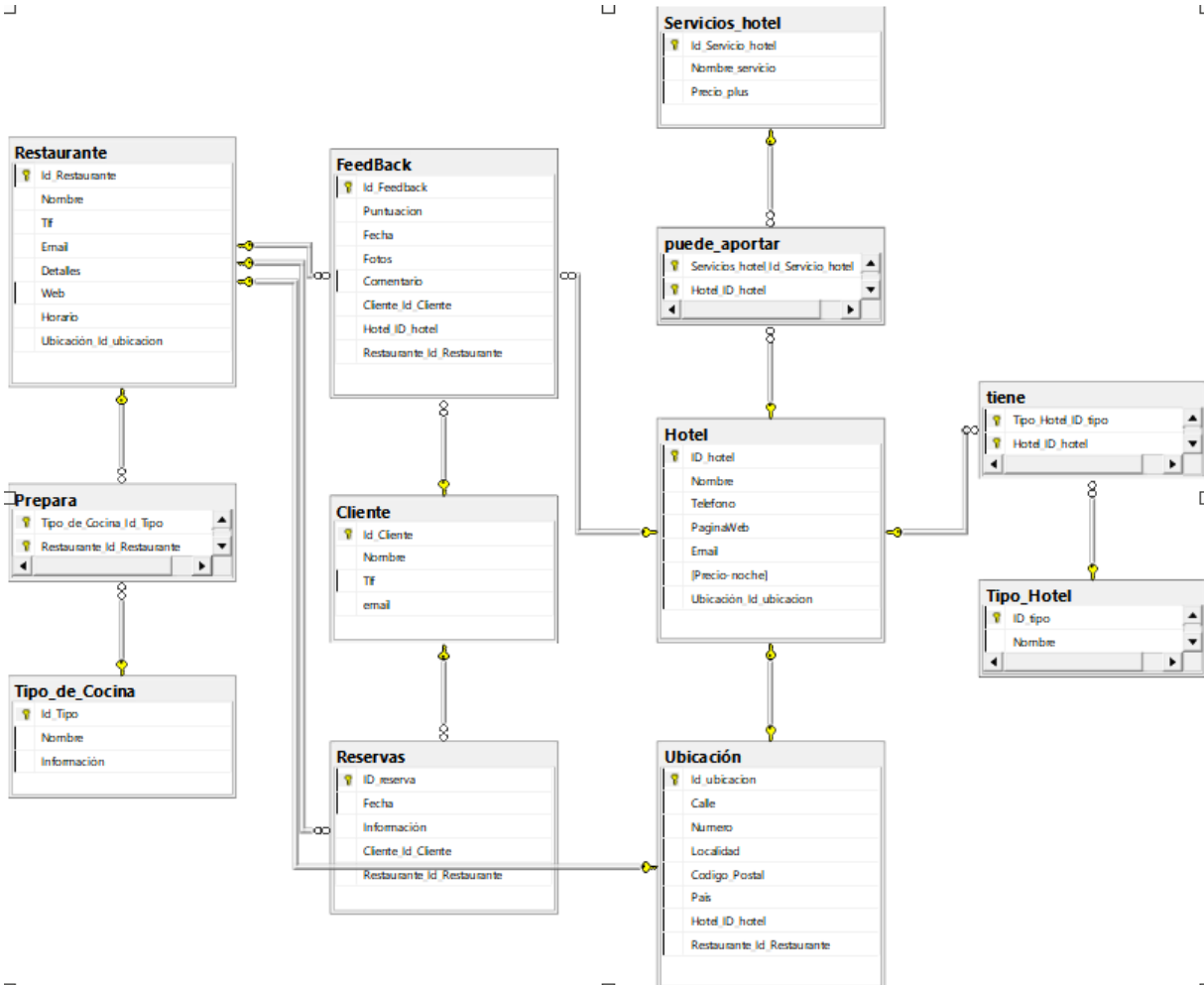
ALTER TABLE tiene
    ADD CONSTRAINT tiene_Tipo_Hotel_FK FOREIGN KEY
    (
        Tipo_Hotel_ID_tipo
    )
    REFERENCES Tipo_Hotel
    (
        ID_tipo
    )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
GO

-- Informe de Resumen de Oracle SQL Developer Data Modeler:
--
-- CREATE TABLE                      12
-- CREATE INDEX                       4
-- ALTER TABLE                      25
-- CREATE VIEW                         0
-- ALTER VIEW                         0
-- CREATE PACKAGE                     0
-- CREATE PACKAGE BODY                 0
-- CREATE PROCEDURE                    0
-- CREATE FUNCTION                     0
-- CREATE TRIGGER                      0
-- ALTER TRIGGER                      0
-- CREATE DATABASE                     0
-- CREATE DEFAULT                      0
-- CREATE INDEX ON VIEW                0
-- CREATE ROLLBACK SEGMENT              0
-- CREATE ROLE                         0
-- CREATE RULE                         0
-- CREATE SCHEMA                       0
-- CREATE SEQUENCE                     0
-- CREATE PARTITION FUNCTION            0
-- CREATE PARTITION SCHEME              0
--
-- DROP DATABASE                       0
--
-- ERRORS                             0
-- WARNINGS

```

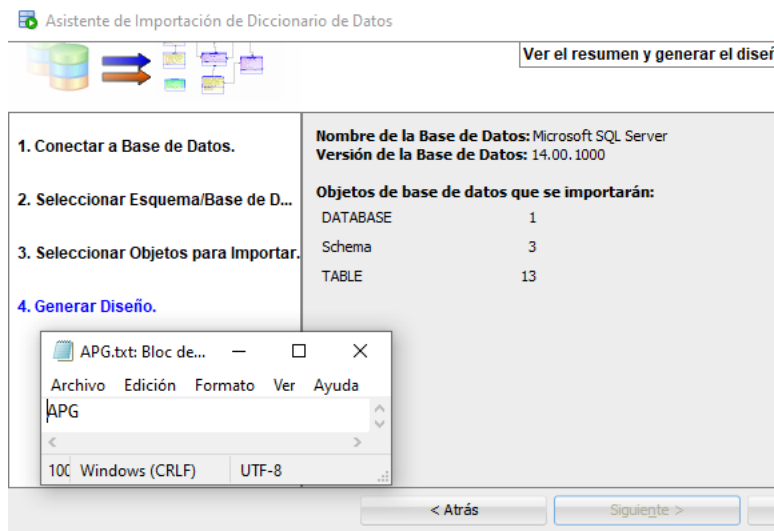
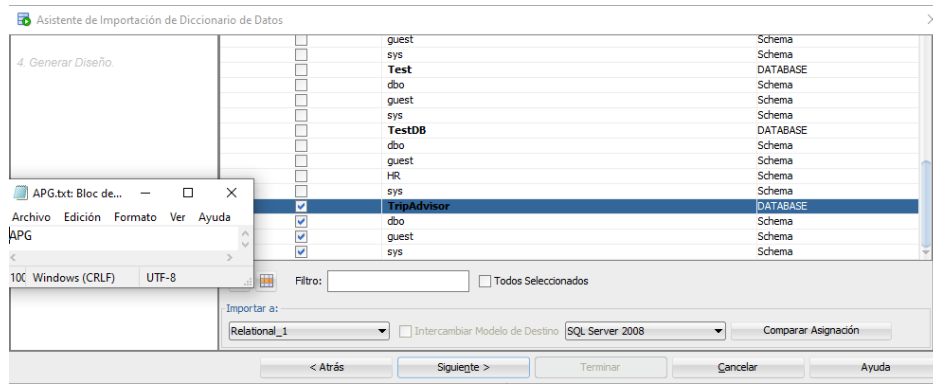


## Modelo Físico



## De SQL Server a Data Modeler (Ingeniería Inversa)

En el programa Oracle Developer, en Archivo-DataModeler-Importar-Diccionario de datos se haría la ingeniería inversa, que permite pasar de SQL Server al modelo Relacional de DataModeler



## ADMINISTRACION DE BASE DE DATOS

### FileTable

Filetable permite almacenar archivos y documentos en tablas especiales de SQL Server denominadas FileTables, para poder tener acceso desde Windows como si estuvieran almacenadas en el sistema de archivos.

Creación de la base de datos de prueba con la que vamos a realizar este apartado.

```
CREATE DATABASE PruebaAPG
ON PRIMARY
(NAME = PruebaAPG_data,
FILENAME = 'C:\FileTable\PruebaAPG.mdf'),
FILEGROUP FileStreamFG

CONTAINS FILESTREAM
(NAME = PruebaAPG,
FILENAME = 'C:\FileTable\PruebaAPG_Container')
```

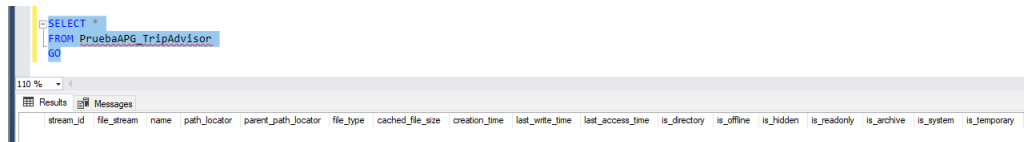
```
LOG ON
(NAME = PruebaAPG_Log,
FILENAME = 'C:\FileTable\PruebaAPG_Log.ldf')

WITH FILESTREAM
(NON_TRANSACTED_ACCESS = FULL,
DIRECTORY_NAME = 'FileTableTripAdvisor');
GO
```

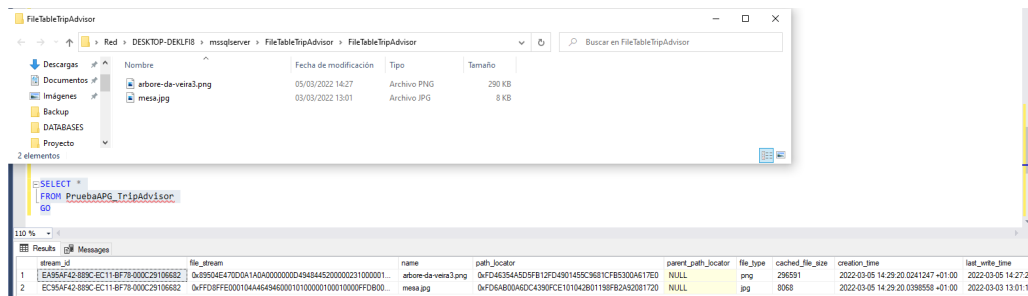
Se crea la tabla con FileTable directorio.

```
use PruebaAPG
CREATE TABLE PruebaAPG_TripAdvisor
AS FILETABLE
WITH
(FileTable_Directory = 'FileTableTripAdvisor',
FileTable_Collate_Filename = database_default);
GO
```

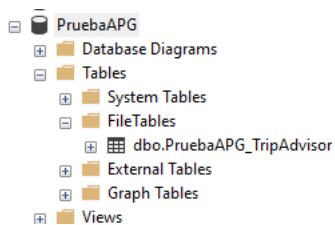
Comprobación antes de introducir archivos en la carpeta



Ejemplo de como se ve un Select al haber introducido mediante Windows las dos imágenes en la carpeta creada de FileTable

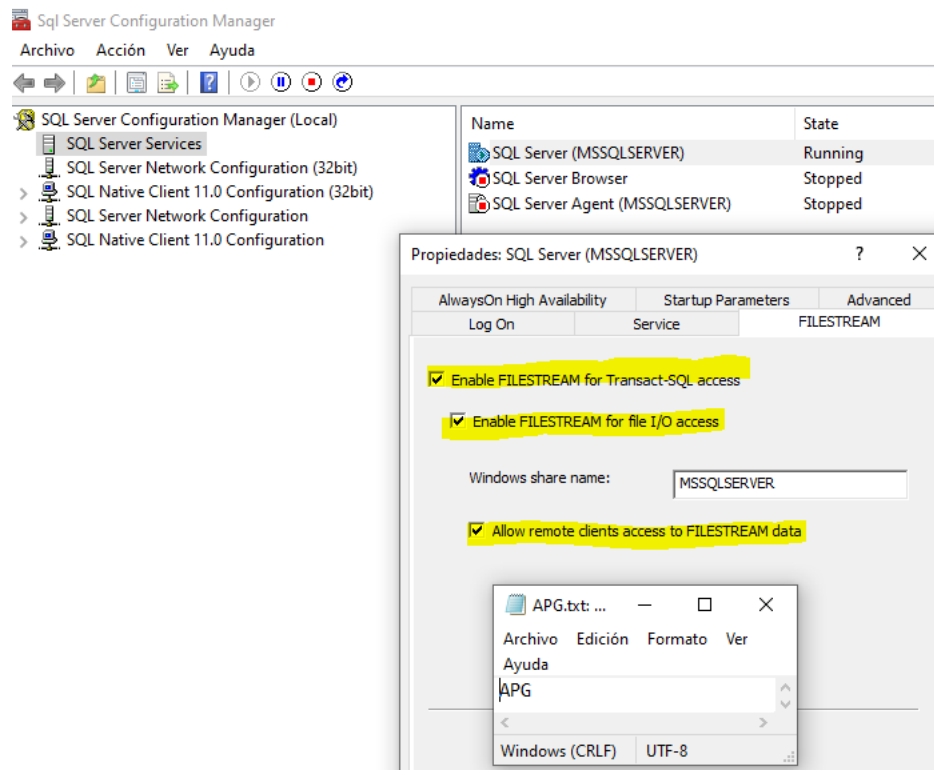


La carpeta Filetable esta dentro de su propia subcarpeta.



## FileStream

Abriendo el Configuration Manager, en propiedades del servidor de SQL Server es necesario realizar el siguiente cambio



A continuación, se ejecutan estas dos sentencias para habilitar el acceso de nivel 2.

```
EXEC sp_configure
go
EXEC sp_configure filestream_access_level
go

EXEC sp_configure filestream_access_level, 2
RECONFIGURE
GO
```

Ahora es necesario modificar la tabla TripAdvisor para generarle un nuevo FileGroup y con un Filestream vinculado.

```
USE TripAdvisor
GO

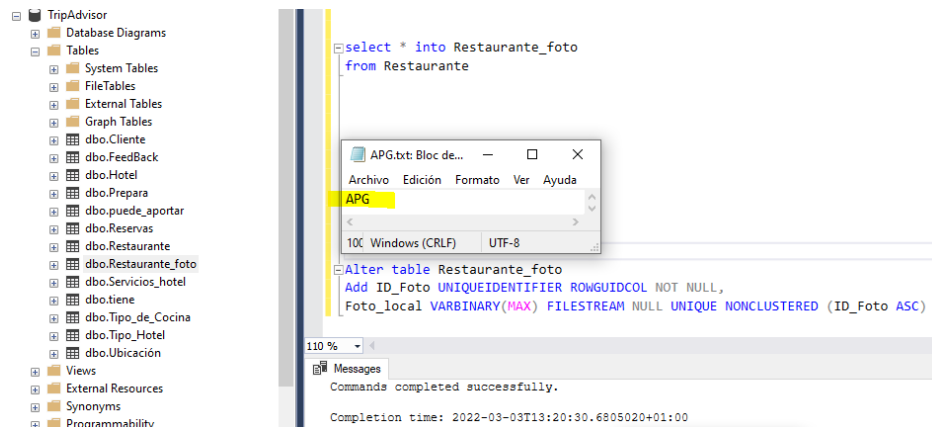
alter database TripAdvisor
add filegroup [FileStream_APG]
contains filestream
go
```

Se copia una tabla de la base de datos a otra nueva para trabajar en ella por si se cometen errores. Además, se modifica dicha tabla.

```
select * into Restaurante_foto
from Restaurante

--SE MODIFICA LA TABLA PARA PERMITIR EL FILESTREAM

Alter table Restaurante_foto
Add ID_Foto UNIQUEIDENTIFIER ROWGUIDCOL NOT NULL,
Foto_local VARBINARY(MAX) FILESTREAM NULL UNIQUE NONCLUSTERED (ID_Foto ASC)
```



Introducimos datos en la tabla. En la columna de **ID\_Foto** se genera automáticamente gracias al "NEWID()", y en la columna Foto\_local, se deja en vacío ya que en una sentencia posterior se modifica para que no sea necesario rellenarlo, y que quede en NULL al no introducir datos

```
insert into Restaurante_foto
values (1,'PlayaClub','620202020','playaclub@hotmail.com','Restaurante a pie de playa','www.playaclub.com','24h','-','-',NEWID(),NULL)

insert into Restaurante_foto
values (2,'Abore da Veira','600500415','areboredaveira@hotmail.com','Restaurante con vistas a la ciudad','www.arboredaveira.com','6h','-','-',N
```

The screenshot shows the 'Results' pane of SQL Server Enterprise Manager. The query 'select \* from Restaurante\_foto' has been executed, and the results are displayed in a table with 10 columns: Id\_Restaurante, Nombre, Tlf, Email, Detalles, Web, Horario, Ubicación\_Id\_ubicacion, ID\_Foto, and Foto\_local. The table contains two rows of data.

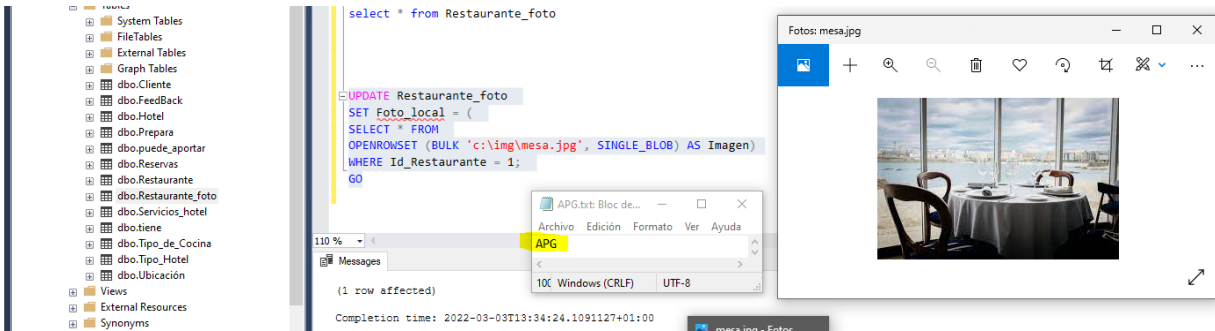
	Id_Restaurante	Nombre	Tlf	Email	Detalles	Web	Horario	Ubicación_Id_ubicacion	ID_Foto	Foto_local
1	1	PlayaClub	620202020	playaclub@hotmail.com	Restaurante a pie de playa	www.playaclub.com	24h	-	511F523D-BB2C-484C-B0A6-0EB4949854FB	NULL
2	2	Abore da Veira	600500415	areboredaveira@hotmail.com	Restaurante con vistas a la ciudad	www.arboredaveira.com	6h	-	273C521C-5B5A-4A19-A333-161AA7CE3428	NULL

Se hace el UPDATE para introducir la foto, que será la que se ve en la captura, ya que con FileStream y en SQL Server nada mas que se ve como un código.

```
UPDATE Restaurante_foto
SET Foto_local = (
SELECT * FROM
OPENROWSET (BULK 'c:\img\mesa.jpg', SINGLE_BLOB) AS Imagen)
WHERE Id_Restaurante = 1;
```

GO

```
UPDATE Restaurante_foto
SET Foto_local = (
SELECT * FROM
OPENROWSET (BULK 'c:\img\arobredaveira.png', SINGLE_BLOB) AS Imagen)
WHERE Id_Restaurante = 2;
GO
```



Resultado final

Id_Restaurante	Nombre	Tlf	Email	Detalles	Web	Horario	Ubicación_Id_ubicacion	ID_Foto	Foto_local
1	PlayaClub	620202020	playaclub@hotmail.com	Restaurante a pie de playa	www.playaclub.com	24h	-	511F523D-BB2C-4B4C-80A6-0EB4949854FB	0xFFD8FFE000104A46494600010100000100010000FFDB00...
2	Albore da Veira	600500415	areboredaveira@hotmail.com	Restaurante con vistas a la ciudad	www.alboredaveira.com	6h	-	273C521C-5B5A-4A19-A333-161AA7CE3428	0x89504E470D0A1A0A00000000D4948445200000231000001...

## Visualización mediante POWER BI

Se crea una nueva tabla con la que trabajar similar a la anterior.

```
drop table if exists Fotos_hotel

CREATE Table Fotos_hotel
(Fotos_hotelId int,
Fotos_hotelName varchar(255),
Fotos_hotelImage varbinary(max))
GO
```

Se introducen datos en la tabla.

```
INSERT INTO dbo.Fotos_hotel (Fotos_hotelId,Fotos_hotelName,Fotos_hotelImage)
SELECT 1,'Blue Hotel',
* FROM OPENROWSET
( BULK 'C:\img\blue.jpg',SINGLE_BLOB) as Imagen
GO

INSERT INTO dbo.Fotos_hotel(Fotos_hotelId,Fotos_hotelName,Fotos_hotelImage)
SELECT 2,'Hotel Finisterre',
* FROM OPENROWSET
( BULK 'C:\img\finisterre.jpg',SINGLE_BLOB) as Imagen
GO
```

```

INSERT INTO dbo.Fotos_hotel(Fotos_hotelId,Fotos_hotelName,Fotos_hotelImage)
SELECT 3,'Trip Maria Pita',
* FROM OPENROWSET
( BULK 'C:\img\trip.jpg',SINGLE_BLOB) as Imagen
go

```

```

SELECT * FROM dbo.Fotos_hotel
GO

```

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'TripAdvisor' database is expanded, showing the 'Fotos\_hotel' table. The table structure is as follows:

Column Name	Data Type	Nullable
Fotos_hotelId	int	no
Fotos_hotelName	varchar(255)	yes
Fotos_hotelImage	varbinary(max)	yes

On the right, a query window shows the following SQL statement:

```

SELECT 3,'Trip Maria Pita',
* FROM OPENROWSET

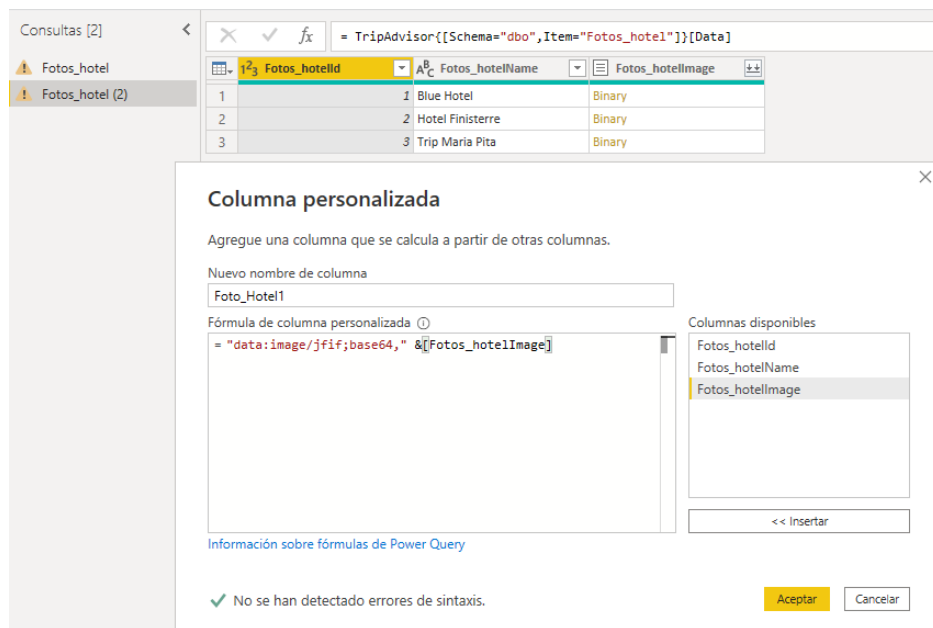
```

Below the query, the 'Results' tab displays the following data:

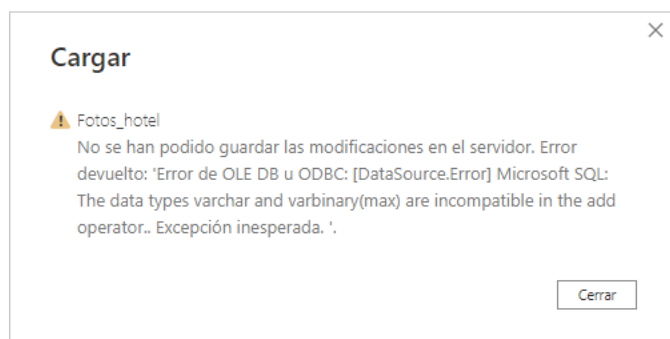
Fotos_hotelId	Fotos_hotelName	Fotos_hotelImage
1	Blue Hotel	0xFFD8FFE000104A46494600010100000100010000FFE100...
2	Hotel Finisterre	0xFFD8FFE000104A46494600010100000100010000FFE100...
3	Trip Maria Pita	0xFFD8FFE000104A46494600010101006400640000FFE100...

The screenshot shows the 'Navegador' (Navigator) window in SQL Server Enterprise Manager. The left pane shows the database structure, with 'Fotos\_hotel' selected under the 'TripAdvisor' database. The right pane shows the 'Fotos\_hotel' table structure:

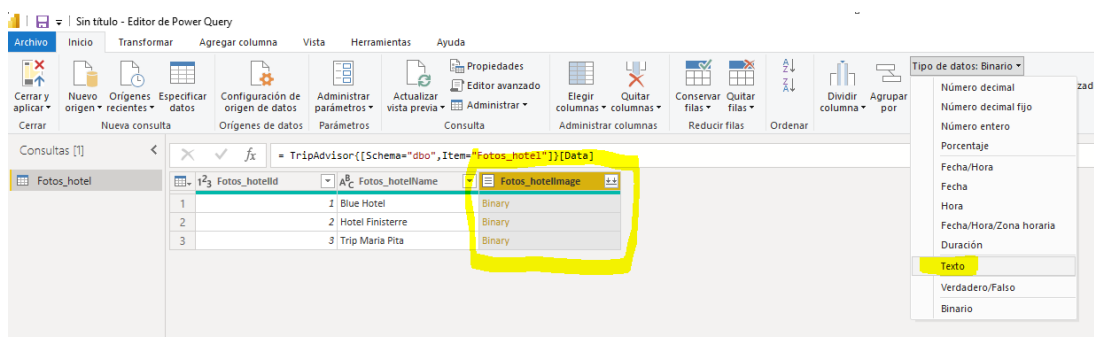
Fotos_hotelId	Fotos_hotelName	Fotos_hotelImage
1	Blue Hotel	Binary
2	Hotel Finisterre	Binary
3	Trip Maria Pita	Binary



Aquí me genera este fallo...

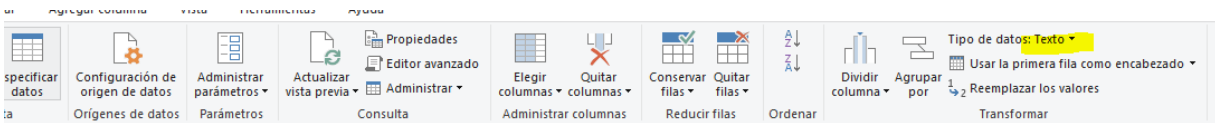


Resulta que para corregir dicho problema hay que cambiar el tipo de datos de la columna de la imagen de binary a text como se muestra en la siguiente imagen.





Con el siguiente resultado....



Transformar

Tipo de datos: **Texto**

Usar la primera fila como encabezado

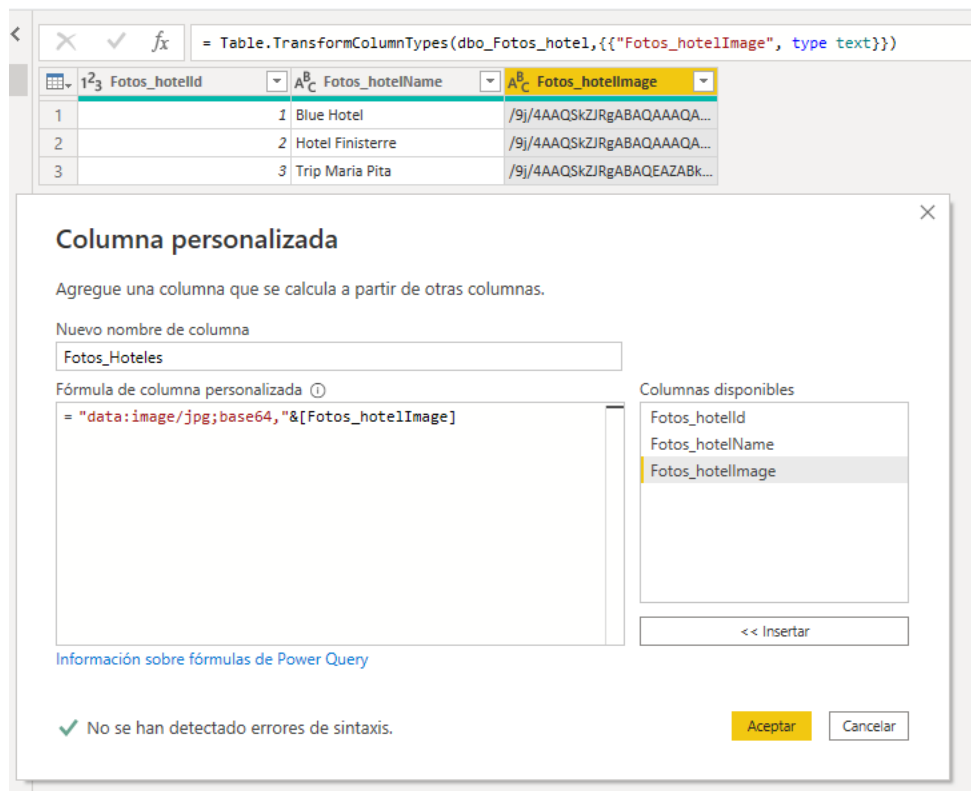
Reemplazar los valores

Transformar

Table.TransformColumnTypes(dbo\_Fotos\_hotel,{{"Fotos\_hotelImage", type text}})

Fotos_hotelId	Fotos_hotelName	Fotos_hotelImage
1	Blue Hotel	/9j/4AAQSkZJRgABAQAAQABAAQ/4QBIRXhpZgAATU0AKgAAAABQESAAAMAA...
2	Hotel Finisterre	/9j/4AAQSkZJRgABAQAAQABAAQ/4QBIRXhpZgAATU0AKgAAAABQESAAAMAA...
3	Trip Maria Pita	/9j/4AAQSkZJRgABAQEAZABkAAD/4QBIRXhpZgAATU0AKgAAAABQESAAAMAA...

Después se añade una columna personalizada con el código que se ve en pantalla



Columna personalizada

Agregue una columna que se calcula a partir de otras columnas.

Nuevo nombre de columna

Fotos\_Hoteles

Fórmula de columna personalizada ⓘ

= "data:image/jpeg;base64," & [Fotos\_hotelImage]

Columnas disponibles

- Fotos\_hotelId
- Fotos\_hotelName
- Fotos\_hotelImage

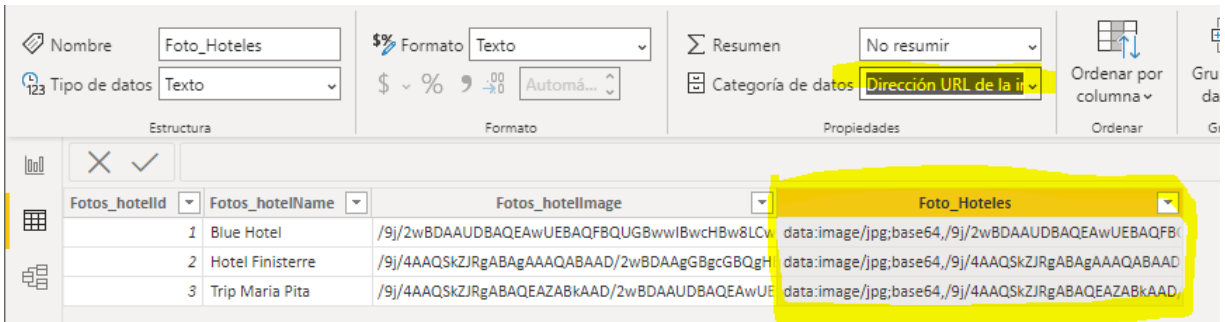
<< Insertar

Información sobre fórmulas de Power Query

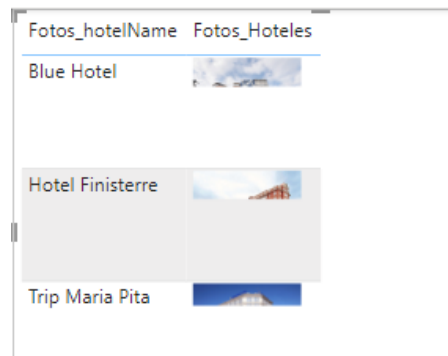
✓ No se han detectado errores de sintaxis.

Aceptar Cancelar

Por ultimo, hay que cambiar la categoría de datos de esa columna a "Dirección URL de imagen"



Finalmente se ven así. No tengo muy claro porque se me ven entrecortadas a pesar de haberlo intentado varias veces.



Para entender bien los procesos, se siguió el siguiente video y la siguiente página.

<https://www.youtube.com/watch?v=Q82yzcfkqAc&t=261s>

#### Complete Guide to Embed Images in Power BI Reports (Part I)

This blog is the first part of embedding images in Power BI. This blog series will discuss almost all methods to embed images in Power BI reports and dashboards. I will try to explain each method's pros and cons with technical limitations, and in the last blog of this series, I will share an intuitive way to embed images without any

<https://www.alphabold.com/embed-images-in-power-bi-reports-part-i/>



## Base de datos Contenida

Lo primero que hay que hacer es activar la siguiente opción

- + Database Snapshots
- + AdventureWorks2017
- + Colegio\_Medico
- + Contenida\_APG
- + GESTOR
- + Northwind
- + PruebaAPG
- + pubs
- + Test
- + TestDB
- + TripAdvisor
- + **TripAdvisorContenida\_APG**
- + WideWorldImporters-Full
- + Security
- + Server Objects
- + Replication
- + PolyBase
- + Always On High Availability
- + Management
- + Integration Services Catalogs
- + SQL Server Agent
- + XEvent Profiler

### Connect to Server

## SQL Server

Login | Connection Properties | Always Encrypted | Additional Connection Parameters

Server

Type the server name, or choose it from the drop-down list.

Server type: Database Engine

Server name:

Authentication: SQL Server Authentication

Login: AdrianPG

Password:

☐ Remember password

Connect Cancel Help Options <<

### Connect to Server

## SQL Server

Login | Connection Properties | Always Encrypted | Additional Connection Parameters

Enter additional connection string parameters (will be sent in clear text):

DATABASE=Contenida\_APG  
DATABASE=TripAdvisorContenida\_APG

(Note: Connection string parameters override graphical selections on other panels)

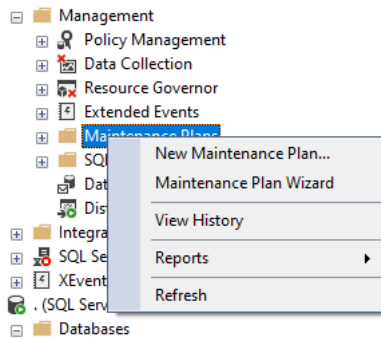
Connect Cancel Help Options <<

- + (SQL Server 14.0.1000.169 - AdrianPG)
- + Databases
- + TripAdvisorContenida\_APG
- + XEvent Profiler

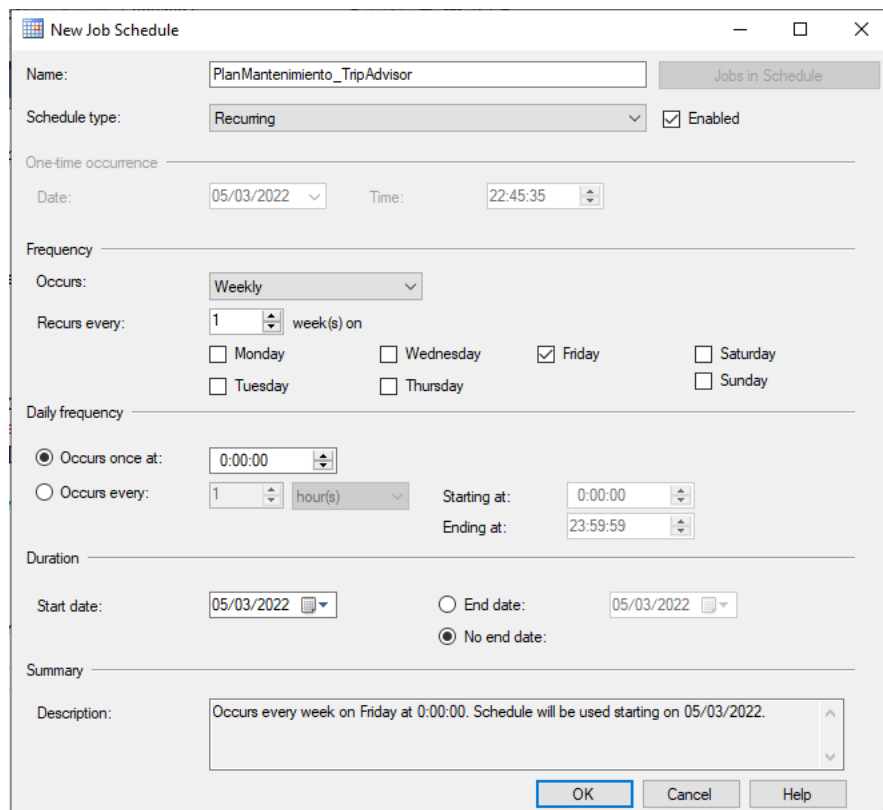
## Planes de Mantenimiento

Los planes de mantenimiento es una función muy útil que permite generar una serie de órdenes para que ejecute ciertas funciones a una o más bases de datos para su correcto orden y mantenimiento.

En mi caso puse de ejemplo que realice un Backup semanalmente de la base de datos TripAdvisor



Establecimiento de los periodos.



**New Job Schedule**

Name: PlanMantenimiento\_TripAdvisor Jobs in Schedule

Schedule type: Recurring ☒ Enabled

One-time occurrence

Date: 05/03/2022 Time: 22:45:35

Frequency

Occurs: Weekly

Recurs every: 1 week(s) on

☐ Monday ☐ Wednesday ☒ Friday ☐ Saturday

☐ Tuesday ☐ Thursday ☐ Sunday

Daily frequency

☒ Occurs once at: 0:00:00

☐ Occurs every: 1 hour(s) Starting at: 0:00:00 Ending at: 23:59:59

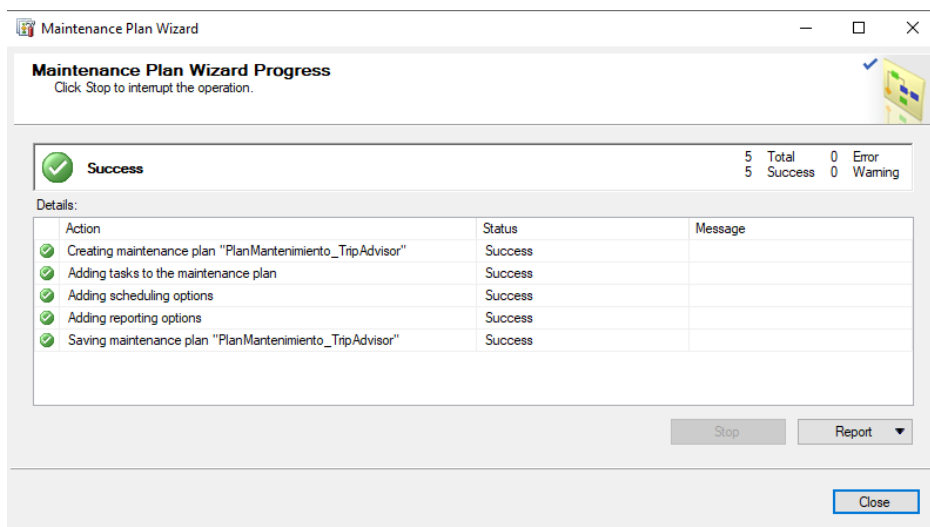
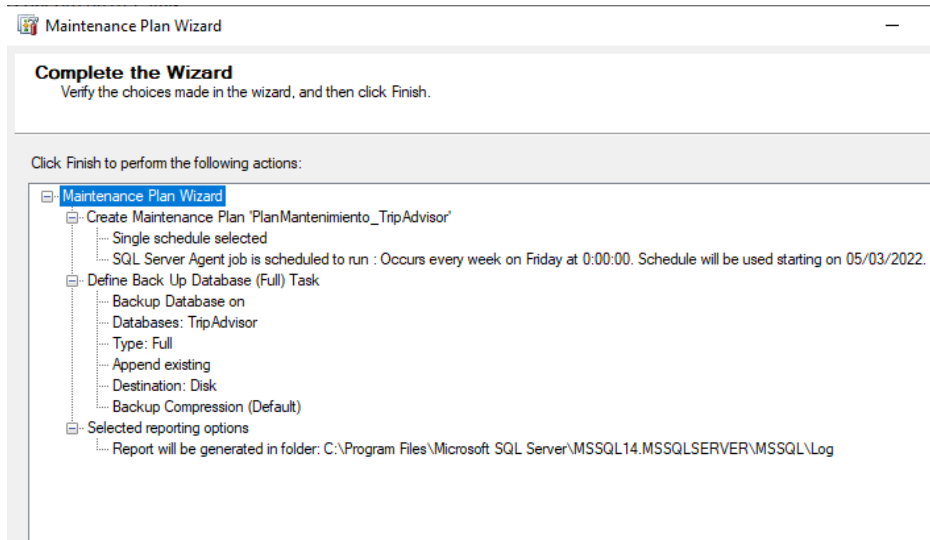
Duration

Start date: 05/03/2022 ☐ End date: 05/03/2022 ☒ No end date:

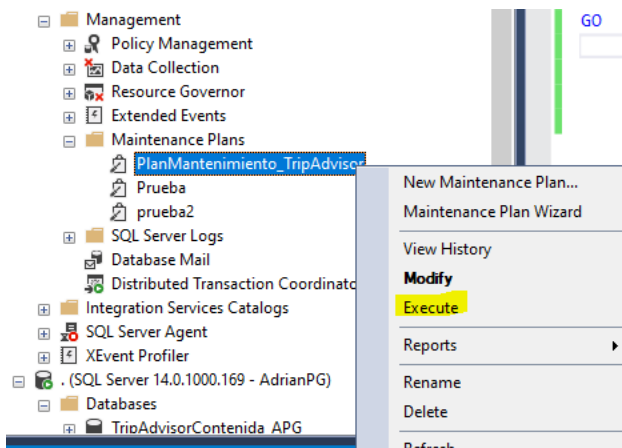
Summary

Description: Occurs every week on Friday at 0:00:00. Schedule will be used starting on 05/03/2022.

OK Cancel Help



En el caso de que se quiera ejecutar al momento sin esperar a que llegue la programación que tenga fijada se hace de la siguiente manera



## Particiones (Split, Merge, Switch, Truncate)

Primero de todo es la creación de una base de datos con la que vamos a trabajar como en ejemplos anteriores.

```
DROP DATABASE IF EXISTS TripAdvisor_FeedbackFecha
GO

CREATE DATABASE TripAdvisor_FeedbackFecha
ON PRIMARY ( NAME = 'TripAdvisor_FeedbackFecha ',
FILENAME = 'C:\Data\TripAdvisor_FeedbackFecha _Fijo.mdf' ,
SIZE = 15360KB , MAXSIZE = UNLIMITED, FILEGROWTH = 0)
LOG ON ( NAME = 'TripAdvisor_FeedbackFecha _log',
FILENAME = 'C:\Data\TripAdvisor_FeedbackFecha _log.ldf' ,
SIZE = 10176KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO

USE TripAdvisor_FeedbackFecha
GO
```

Creación de Filegroups con las valoraciones por fechas

```
ALTER DATABASE [TripAdvisor_FeedbackFecha ] ADD FILEGROUP [APG_Archivo]
GO
ALTER DATABASE [TripAdvisor_FeedbackFecha ] ADD FILEGROUP [APG_2020]
GO
ALTER DATABASE [TripAdvisor_FeedbackFecha ] ADD FILEGROUP [APG_2021]
GO
ALTER DATABASE [TripAdvisor_FeedbackFecha ] ADD FILEGROUP [APG_2022]
GO
```

Consulta...

```
select * from sys.filegroups
GO
```

	name	data_space_id	type	type_desc	is_default	is_system	filegroup_guid	log_filegroup_id	is_read_only	is_autogrow_all_files
1	PRIMARY	1	FG	ROWS_FILEGROUP	1	0	NULL	NULL	0	0
2	APG_Archivo	2	FG	ROWS_FILEGROUP	0	0	765609FD-A5C8-45C3-9384-5BAFCE2D91BB	NULL	0	0
3	APG_2020	3	FG	ROWS_FILEGROUP	0	0	023E8C27-8A2A-4E6E-82CD-8C82014E14EC	NULL	0	0
4	APG_2021	4	FG	ROWS_FILEGROUP	0	0	7FAC8D88-F548-4A51-9098-F83621871C1D	NULL	0	0
5	APG_2022	5	FG	ROWS_FILEGROUP	0	0	E451CF9C-89E8-41B6-AE96-43E5C82EBB65	NULL	0	0

Query executed successfully. | (local) (14.0 RTM) | DESKTOP-DEKLF18\APG (67) | TripAdvisor\_FeedbackFecha | 00:00:00 | 5 rows

Creación de los archivos contenedores de las particiones

```
ALTER DATABASE [TripAdvisor_FeedbackFecha]
ADD FILE ( NAME = 'Fechas_Archivo', FILENAME = 'c:\DATA\Fechas_Archivo.ndf', SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 2MB )
TO FILEGROUP [APG_Archivo]
GO
```

```

ALTER DATABASE [TripAdvisor_FeedbackFecha]
ADD FILE ( NAME = 'Fechas_2020', FILENAME = 'c:\DATA\Fechas_2020.ndf', SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 2MB )
TO FILEGROUP [APG_2020]
GO

ALTER DATABASE [TripAdvisor_FeedbackFecha]
ADD FILE ( NAME = 'Fechas_2021', FILENAME = 'c:\DATA\Fechas_2021.ndf', SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 2MB )
TO FILEGROUP [APG_2021]
GO

ALTER DATABASE [TripAdvisor_FeedbackFecha]
ADD FILE ( NAME = 'Fechas_2022', FILENAME = 'c:\DATA\Fechas_2022.ndf', SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 2MB )
TO FILEGROUP [APG_2022]
GO

```

Consulta de nuevo de los archivos y de los filegroups

```

select * from sys.filegroups
GO

select * from sys.database_files
GO

```

	name	data_space_id	type	type_desc	is_default	is_system	filegroup_guid	log_filegroup_id	is_read_only	is_autogrow_all_files
1	PRIMARY	1	FG	ROWS_FILEGROUP	1	0	NULL	NULL	0	0
2	APG_Archivo	2	FG	ROWS_FILEGROUP	0	0	765609FD-A5C8-45C3-9384-5BAFCE2D91BB	NULL	0	0
3	APG_2020	3	FG	ROWS_FILEGROUP	0	0	023E8C27-8A2A-4E6E-82CD-8C82014E14EC	NULL	0	0
4	APG_2021	4	FG	ROWS_FILEGROUP	0	0	7FAC8D88-F548-4A51-9098-F83621871C1D	NULL	0	0
5	APG_2022	5	FG	ROWS_FILEGROUP	0	0	E451CF9C-89E8-41B6-AE96-43E5C82EBB65	NULL	0	0

	file_id	file_guid	type	type_desc	data_space_id	name	physical_name	state	state_desc
1	1	0239C43F-90B9-4D8D-BC52-A822E9A38E43	0	ROWS	1	TripAdvisor_FeedbackFecha	C:\Data\TripAdvisor_FeedbackFecha_Fijo.mdf	0	ONLINE
2	2	06A6342D-4C65-41E1-9B93-5409A0EAF268	1	LOG	0	TripAdvisor_FeedbackFec...	C:\Data\TripAdvisor_FeedbackFecha_log.ldf	0	ONLINE
3	3	2AF3C7DA-8B81-4836-AA90-EB84E9DBFD...	0	ROWS	2	Fechas_Archivo	c:\DATA\Fechas_Archivo.ndf	0	ONLINE
4	4	89CE6871-1E34-45BC-A4D5-877572623859	0	ROWS	3	Fechas_2020	c:\DATA\Fechas_2020.ndf	0	ONLINE
5	5	ED9BB51B-97DF-4E83-B67C-790034A8E62F	0	ROWS	4	Fechas_2021	c:\DATA\Fechas_2021.ndf	0	ONLINE
6	6	2C462DD8-64EA-4A0E-9379-8AAC1E90EB...	0	ROWS	5	Fechas_2022	c:\DATA\Fechas_2022.ndf	0	ONLINE

## Límites de las funciones

- Creación de la función donde se establecen los límites de fechas.
  - 1. Datos de 2020
  - 2. Datos de 2020
  - 3. Datos posteriores a 01/01/2021

```

CREATE PARTITION FUNCTION FN_fecha_feedback (datetime)
AS RANGE RIGHT
FOR VALUES ( '2020-01-01', '2021-01-01' )
GO

```

Ahora se le asignan los rangos (partition scheme)

```

create partition scheme feedback_fecha
as partition FN_fecha_feedback
to (APG_Archivo, APG_2020, APG_2021, APG_2022)
go

```

Se crea una tabla “dbo.FeedbackAPG” de ejemplo dentro de la base de datos

```

drop table if exists dbo.FeedbackAPG
go

create table dbo.FeedbackAPG(
    id_FeedBack int not null,
    dni_cliente varchar(9),
    fecha datetime)
on feedback_fecha --scheme
(fecha)--columna a aplicar el schema
go

```

Insertamos datos en los diferentes años

```

--Datos del 2019:
insert into dbo.FeedbackAPG
values (1, '53155430R', '2019-01-11 10:00:00'),
      (2, '52754615T', '2019-01-14 10:30:00'),
      (3, '53166530Y', '2019-01-19 10:45:00')
go

--Datos del 2020:
insert into dbo.FeedbackAPG
values (4, '32429452G', '2020-01-2 09:30:00'),
      (5, '53166533G', '2020-01-20 09:30:00'),
      (6, '53166530Y', '2020-01-23 09:45:00')
go

--Datos del 2021:
insert into dbo.FeedbackAPG
values (7, '47367813A', '2021-01-15 09:30:00' ),
      (8, '41565300A', '2021-01-25 09:30:00' ),
      (9, '53166530Y', '2021-01-8 09:45:00')
go

--Datos del 2022:
insert into dbo.FeedbackAPG
values (10, '53166530y', '2022-01-28 09:30:00'),
      (11, '47367813A', '2022-02-03 09:35:00'),
      (12, '63353362D', '2022-02-03 09:40:00')
go

```



```
--CONSULTA

select * from dbo.FeedbackAPG
go
```

	id_FeedBack	dni_cliente	fecha
1	1	53155430R	2019-01-11 10:00:00.000
2	2	52754615T	2019-01-14 10:30:00.000
3	3	53166530Y	2019-01-19 10:45:00.000
4	4	32429452G	2020-01-02 09:30:00.000
5	5	53166533G	2020-01-20 09:30:00.000
6	6	53166530Y	2020-01-23 09:45:00.000
7	7	47367813A	2021-01-15 09:30:00.000
8	8	41565300A	2021-01-25 09:30:00.000
9	9	53166530Y	2021-01-08 09:45:00.000
10	10	53166530y	2022-01-28 09:30:00.000
11	11	47367813A	2022-02-03 09:35:00.000
12	12	63353362D	2022-02-03 09:40:00.000

- Select para ver las diferentes particiones

```
select name, create_date, value from sys.partition_functions f
inner join sys.partition_range_values rv
on f.function_id=rv.function_id
where f.name = 'FN_fecha_feedback'
go
```

	name	create_date	value
1	FN_fecha_feedback	2022-03-06 18:57:11.800	2020-01-01 00:00:00.000
2	FN_fecha_feedback	2022-03-06 18:57:11.800	2021-01-01 00:00:00.000

- Select para ver cuantos registros hay en cada partición

```
select p.partition_number, p.rows from sys.partitions p
inner join sys.tables t
on p.object_id=t.object_id and t.name = 'FeedbackAPG'
GO
```

	partition_number	rows
1	1	3
2	2	3
3	3	6

- Particiones, Filegroups y Limites

```
DECLARE @TableName NVARCHAR(200) = N'FeedbackAPG'
SELECT SCHEMA_NAME(o.schema_id) + '.' + OBJECT_NAME(i.object_id) AS [object] ,
p.partition_number AS [p#] ,
fg.name AS [filegroup]
, p.rows ,
au.total_pages AS pages ,
CASE boundary_value_on_right WHEN 1 THEN 'less than' ELSE 'less than or equal to' END as comparison ,
rv.value ,
CONVERT (VARCHAR(6), CONVERT (INT, SUBSTRING (au.first_page, 6, 1) +
SUBSTRING (au.first_page, 5, 1))) + ':' + CONVERT (VARCHAR(20),
CONVERT (INT, SUBSTRING (au.first_page, 4, 1) + SUBSTRING (au.first_page, 3, 1) +
SUBSTRING (au.first_page, 2, 1) + SUBSTRING (au.first_page, 1, 1))) AS
first_page FROM sys.partitions p INNER JOIN sys.indexes i ON p.object_id = i.object_id AND p.index_id = i.index_id INNER JOIN sys.objects o
ON p.object_id = o.object_id INNER JOIN sys.system_internals_allocation_units au ON p.partition_id = au.container_id
INNER JOIN sys.partition_schemes ps ON ps.data_space_id = i.data_space_id INNER JOIN sys.partition_functions f ON f.function_id = ps.function_id
INNER JOIN sys.destination_data_spaces dds ON dds.partition_scheme_id = ps.data_space_id AND dds.destination_id = p.partition_number
INNER JOIN sys.filegroups fg ON dds.data_space_id = fg.data_space_id LEFT OUTER JOIN sys.partition_range_values rv ON f.function_id = rv.function_id
AND p.partition_number = rv.boundary_id WHERE i.index_id < 2 AND o.object_id = OBJECT_ID(@TableName);
GO
```

	object	p#	filegroup	rows	pages	comparison	value	first_page
1	dbo.FeedbackAPG	1	APG_Archivo	3	9	less than	2020-01-01 00:00:00.000	3:8
2	dbo.FeedbackAPG	2	APG_2020	3	9	less than	2021-01-01 00:00:00.000	4:8
3	dbo.FeedbackAPG	3	APG_2021	6	9	less than	NULL	5:8

## Split

Para dividir una partición en otra a mayores.

En este caso, los Feedback posteriores al 01/01/2022 se irán a otra partición (4)

```
ALTER PARTITION FUNCTION FN_fecha_feedback()
    SPLIT RANGE ('2022-01-01');
GO
```

## Consulta:

```
select *, $Partition.FN_fecha_feedback(fecha) AS partition
from dbo.FeedbackAPG
go
```

	id_FeedBack	dni_cliente	fecha	partition
1	1	53155430R	2019-01-11 10:00:00.000	1
2	2	52754615T	2019-01-14 10:30:00.000	1
3	3	53166530Y	2019-01-19 10:45:00.000	1
4	4	32429452G	2020-01-02 09:30:00.000	2
5	5	53166533G	2020-01-20 09:30:00.000	2
6	6	53166530Y	2020-01-23 09:45:00.000	2
7	7	47367813A	2021-01-15 09:30:00.000	3
8	8	41565300A	2021-01-25 09:30:00.000	3
9	9	53166530Y	2021-01-08 09:45:00.000	3
10	10	53166530y	2022-01-28 09:30:00.000	4
11	11	47367813A	2022-02-03 09:35:00.000	4
12	12	63353362D	2022-02-03 09:40:00.000	4

	object	p#	filegroup	rows	pages	comparison	value	first_page
1	dbo.FeedbackAPG	1	APG_Archivo	3	9	less than	2020-01-01 00:00:00.000	3:8
2	dbo.FeedbackAPG	2	APG_2020	3	9	less than	2021-01-01 00:00:00.000	4:8
3	dbo.FeedbackAPG	3	APG_2021	3	9	less than	2022-01-01 00:00:00.000	5:8
4	dbo.FeedbackAPG	4	APG_2022	3	9	less than	NULL	6:8

## Merge

Para fusionar dos particiones. En este caso se fusionan las que anteriormente había separado

```
alter partition function FN_fecha_feedback ()
merge range ('2020-01-01')
go
```

	object	p#	filegroup	rows	pages	comparison	value	first_page
1	dbo.FeedbackAPG	1	APG_Archivo	6	9	less than	2021-01-01 00:00:00.000	3:8
2	dbo.FeedbackAPG	2	APG_2021	3	9	less than	2022-01-01 00:00:00.000	5:8
3	dbo.FeedbackAPG	3	APG_2022	3	9	less than	NULL	6:8

## Switch

Switch sirve para pasar el contenido de una partición a otra partición.

En este ejemplo, una vez más, se crea una tabla de pruebas.

```
create table dbo.OpinionesAPG(
    id_FeedBack int not null,
    dni_cliente varchar(9),
    fecha datetime)
on APG_Archivo
go
```

Transferimos los registros desde Feedback a Opiniones.

```
alter table dbo.FeedbackAPG
switch partition 1 to OpinionesAPG
```

go

Consulta:

```
--Consulta de las dos tablas
select * from FeedbackAPG
go
select * from OpinionesAPG
go
```

	id_FeedBack	dni_cliente	fecha
1	7	47367813A	2021-01-15 09:30:00.000
2	8	41565300A	2021-01-25 09:30:00.000
3	9	53166530Y	2021-01-08 09:45:00.000
4	10	53166530y	2022-01-28 09:30:00.000
5	11	47367813A	2022-02-03 09:35:00.000
6	12	63353362D	2022-02-03 09:40:00.000

	id_FeedBack	dni_cliente	fecha
1	1	53155430R	2019-01-11 10:00:00.000
2	2	52754615T	2019-01-14 10:30:00.000
3	3	53166530Y	2019-01-19 10:45:00.000
4	4	32429452G	2020-01-02 09:30:00.000
5	5	53166533G	2020-01-20 09:30:00.000
6	6	53166530Y	2020-01-23 09:45:00.000

Query executed successfully. (local) (14.0 RTM) DESKTOP

Con el select completo de particiones, filegroup, registros y limites

	object	p#	filegroup	rows	pages	comparison	value	first_page
1	dbo.FeedbackAPG	1	APG_Archivo	0	0	less than	2021-01-01 00:00:00.000	0:0
2	dbo.FeedbackAPG	2	APG_2021	3	9	less than	2022-01-01 00:00:00.000	5:8
3	dbo.FeedbackAPG	3	APG_2022	3	9	less than	NULL	6:8

## Truncate

Sirve para borrar el contenido de las particiones. Se borra la partición 3.

```
TRUNCATE TABLE FeedbackAPG
WITH (PARTITIONS (3));
go
```

select \* from FeedbackAPG  
GO

Results			
	id_FeedBack	dni_cliente	fecha
1	7	47367813A	2021-01-15 09:30:00.000
2	8	41565300A	2021-01-25 09:30:00.000
3	9	53166530Y	2021-01-08 09:45:00.000

Results								
	object	p#	filegroup	rows	pages	comparison	value	first_page
1	dbo.FeedbackAPG	1	APG_Archivo	0	0	less than	2021-01-01 00:00:00.000	0:0
2	dbo.FeedbackAPG	2	APG_2021	3	9	less than	2022-01-01 00:00:00.000	5:8
3	dbo.FeedbackAPG	3	APG_2022	0	0	less than	NULL	0:0

## Tablas temporales del sistema

Las tablas temporales del sistema permite generar un histórico de movimientos de una tabla para poder recuperar información en caso de haberlo hecho por error. Es útil también para comparar históricos de movimientos.

Primero de todo es la creación de la BD de ejemplo TripAdv\_TableTemp

```
DROP DATABASE IF EXISTS TripAdv_TableTemp
GO
CREATE DATABASE TripAdv_TableTemp
ON PRIMARY ( NAME = 'TripAdv_TableTemp',
FILENAME = 'C:\Data\TripAdv_TableTemp_Fijo.mdf' ,
SIZE = 15360KB , MAXSIZE = UNLIMITED, FILEGROWTH = 0)
LOG ON ( NAME = 'TripAdv_TableTemp_log',
FILENAME = 'C:\Data\TripAdv_TableTemp_log.ldf' ,
SIZE = 10176KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO
```

Seguimos con la creación de la tabla

```
drop table if exists reservaAPG
go
create table reserva
( id_reserva int primary key clustered,
name_hotel varchar(90) not null,
cliente varchar(90) not null,
informacion varchar(120),
SysStartTime datetime2 generated always as row start not null,
SysEndTime datetime2 generated always as row end not null,
period for System_time (SysStartTime,SysEndTime) )
with (System_Versioning = ON (History_Table = dbo.reservaAPG_historico)
)
go
```

Ahora insertamos valores a la tabla. Hay que especificar que columnas queremos realizar inserciones ya que si no daría fallo, ya que hay dos columnas (*SysStartTime* y *SysEndTime*) que no es necesario introducir valor.

```
insert into reservaAPG ([id_reserva],[name_hotel],[cliente],[informacion])
values ( 1, 'Maria Pita', 'David', 'Preferencia cama individual'),
      ( 2, 'Blue Hotel', 'Adrian', 'Preferencia dos camas individuales'),
      ( 3, 'Hotel Plaza', 'Pardiñas', 'Cama doble si puede ser'),
      ( 4, 'Finisterre', 'Ana', 'Habitacion con jacuzzi'),
      ( 5, 'Avenida', 'Pedro', 'Habitacion con terraza'),
      ( 6, 'Royale', 'Juan', 'Habitacion fumadores')
go
```

Se realiza una consulta de la tabla con los datos introducidos hasta el momento

```
--Consulta
select * from reservaAPG
go
```

	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
2	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
3	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
6	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999

La tabla de históricos, al no haber ningún movimiento, no hay datos...

```
select * from dbo.reservaAPG_historico
```

id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
------------	------------	---------	-------------	--------------	------------

Empezamos haciendo **modificaciones** (update, borrado y nuevas inserciones). Lo ideal sería ir poco a poco haciendo dichas modificaciones para que los cambios se vean mas claros.

## Update

Modificación de un campo de una inserción

```
update reservaAPG
set informacion = 'Cama doble'
where id_reservaAPG = 2
GO
```

Results		Messages				
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
2	2	Blue Hotel	Adrian	Cama doble	2022-03-07 13:42:08.7814794	9999-12-31 23:59:59.9999999
3	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
6	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	2022-03-07 13:42:08.7814794

Consulta + Histórico

## Borrado

Procedemos a borrar un registro completo.

```
delete from reservaAPG
where id_reservaAPG=1
go
```

Results		Messages				
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Cama doble	2022-03-07 13:42:08.7814794	9999-12-31 23:59:59.9999999
2	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
3	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	2022-03-07 13:42:08.7814794
2	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	2022-03-07 13:50:25.8751853

Consulta + Histórico

## Insert

Ahora se inserta una nueva fila completa.

```
insert into reservaAPG ([id_reserva],[name_hotel],[cliente],[informacion])
values ( 7, 'Attica', 'Tomás', 'Suite Premium')
go
```

Results

Messages

	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Cama doble	2022-03-07 13:42:08.7814794	9999-12-31 23:59:59.9999999
2	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
3	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
6	7	Attica	Tomás	Suite Premium	2022-03-07 13:56:10.7972677	9999-12-31 23:59:59.9999999

	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	2022-03-07 13:42:08.7814794
2	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	2022-03-07 13:50:25.8751853

Consulta + Histórico

## Consultas:

- Consulta de todos los movimientos realizados en la tabla

```
select *
from reservaAPG
for system_time all
go
```

Results		Messages				
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Cama doble	2022-03-07 13:42:08.7814794	9999-12-31 23:59:59.9999999
2	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
3	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
6	7	Attica	Tomás	Suite Premium	2022-03-07 13:56:10.7972677	9999-12-31 23:59:59.9999999
7	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	2022-03-07 13:42:08.7814794
8	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	2022-03-07 13:50:25.8751853

- Consulta de un periodo

```
select *
from reservaAPG
for system_time from '2022-03-07 13:42:08.7814794' to '2022-03-07 13:56:10.7972677'
go
```

Results		Messages				
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Cama doble	2022-03-07 13:42:08.7814794	9999-12-31 23:59:59.9999999
2	3	Hotel Plaza	Pardiñas	Cama doble si puede ser	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
3	4	Finisterre	Ana	Habitacion con jacuzzi	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
4	5	Avenida	Pedro	Habitacion con terraza	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
5	6	Royale	Juan	Habitacion fumadores	2022-03-07 13:37:27.9533807	9999-12-31 23:59:59.9999999
6	1	Maria Pita	David	Preferencia cama individual	2022-03-07 13:37:27.9533807	2022-03-07 13:50:25.8751853

- Registros modificados en un periodo de tiempo concreto



```
select *
from reservaAPG
for system_time contained in ('2022-03-07 13:37:27.9533807','2022-03-07 13:42:08.7814794')
GO
```

Results		Messages				
	id_reserva	name_hotel	cliente	informacion	SysStartTime	SysEndTime
1	2	Blue Hotel	Adrian	Preferencia dos camas individuales	2022-03-07 13:37:27.9533807	2022-03-07 13:42:08.7814794

## Tablas en memoria

Las tablas en memoria son tablas que cumplen ACID (atomicidad, consistencia, aislamiento y durabilidad). Se usan para optimizar el rendimiento en el procesamiento de transacciones.

Para crear tablas In Memory es necesario un filegroup Memory\_Optimized y un archivo en su interior.

En este apartado, se creará la base de datos TripAdvAPG\_TBInmemory.

```
drop database if exists TripAdvAPG_TBInmemory
go

create database TripAdvAPG_TBInmemory
go
```

Es necesario activar el "memory\_optimized\_elevate\_to\_snapshot"

```
alter database current
set memory_optimized_elevate_to_snapshot = on
go
```

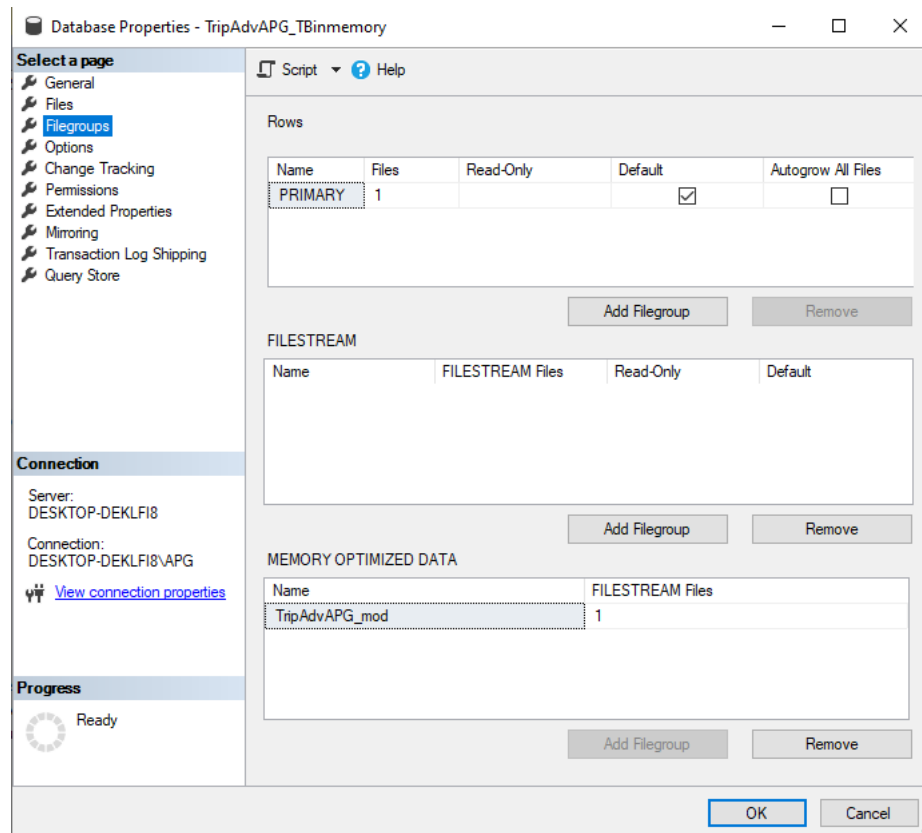
Ahora se modifica la tabla y se añade el filegroup optimizado

```
alter database TripAdvAPG_TBInmemory
add filegroup TripAdvAPG_mod
contains memory_optimized_data
go
```

Se añaden como mínimo un contenedor al filegroup

```
alter database TripAdvAPG_TBInmemory
add file (name='TripAdvAPG_mod1',
filename='c:\data\TripAdvAPG_mod1')
to filegroup TripAdvAPG
go
```

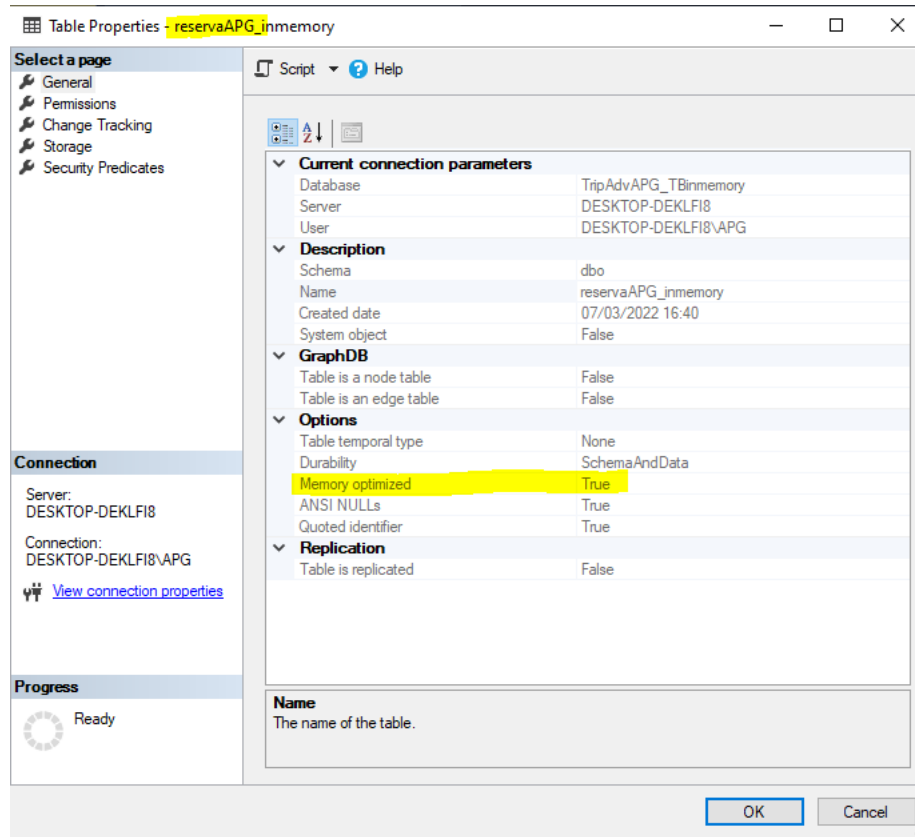
En esta captura se puede apreciar que en las propiedades de la base de datos, en la sección Filegroups se ve abajo la nueva Optimized\_Data



Creación de la tabla. Es necesario que la PK sea "clustered".

```
drop table if exists reservaAPG_inmemory
go
create table reservaAPG_inmemory
(id_reserva int primary key nonclustered, --Es necesario que la PK sea clustered
name_hotel varchar(90) not null,
cliente varchar(90) not null,
informacion varchar(120))
with
(memory_optimized = on,
durability = schema_and_data)
go
```

Ahora en la tabla se puede apreciar como también tiene activado la memoria optimizada.



## GIT

Para dar los primeros pasos en GIT he empleado este tutorial.

### Primeros pasos con git y GitHub

Git es un sistema de control de versiones muy potente y sin duda el más usado en la actualidad que nos permite llevar un registro histórico de todos los cambios realizados en cada uno de los archivos a lo largo del desarrollo de un proyecto.

✂ <https://styde.net/primeros-pasos-con-git-y-github/>



GIT tiene distintas funcionalidades como pueden ser:

- Organización de código. Repositorio de almacenaje de código.
- Controlar versiones. Se puede volver atrás en el código, deshaciendo cambios y volviendo atrás.
- Trabajo en equipo. Mantener varias personas usando el mismo código.



#### Comandos básicos de GIT

- `git init` → crea un repositorio de GIT vacío que gestionará el propio GIT.
- `git status` → Consulta los ficheros que tenemos en el repositorio.
- `git add -A` → Comando para pasar todos los archivos en “espera” del Working Directory al Index / Staging.
- `git add “nombre_fichero”` → La otra manera de pasar del Working Directory al Index Area es así, moviendo lo que interesa.
- `git commit` → “Mueves” (confirmas) pasar de Index / Staging a Git Repository. Un commit es técnicamente un snapshot (instantánea) de cómo está mi directorio de trabajo. Si yo sigo avanzando, siempre puedo volver atrás.
- `git diff` → Sirve para ver los cambios que se han hecho, para comparar.

## Creación de la carpeta y del repositorio

Lo primero es crear una carpeta que sera el directorio del repositorio. En mi caso la cree en C:/

Confirmación de la carpeta `APG_repositorio` creada como carpeta GIT

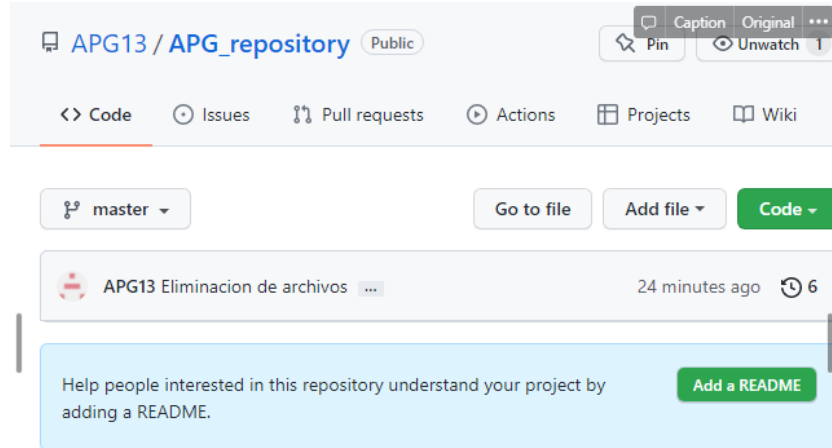
```
APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio
$ git init
Initialized empty Git repository in C:/APG_repositorio/.git/

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git remote add origin https://github.com/APG13/APG_repositorio.git

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git pull origin master
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 17 (delta 2), reused 17 (delta 2), pack-reused 0
Unpacking objects: 100% (17/17), 362.55 KiB | 894.00 KiB/s, done.
From https://github.com/APG13/APG_repositorio
* branch      master      -> FETCH_HEAD
* [new branch] master      -> origin/master

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Situación del repositorio previo a la subida de archivos



Subida de tres archivos con su posterior commit (confirmación)

```
APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Logicol.pdf
    Proyecto1.sql
    Relational_1.pdf

nothing added to commit but untracked files present (use "git add" to track)

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git add -A

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git commit -m "Subida de archivos de modelado"
[master edd9273] Subida de archivos de modelado
3 files changed, 433 insertions(+)
create mode 100644 Logicol.pdf
create mode 100644 Proyecto1.sql
create mode 100644 Relational_1.pdf
```

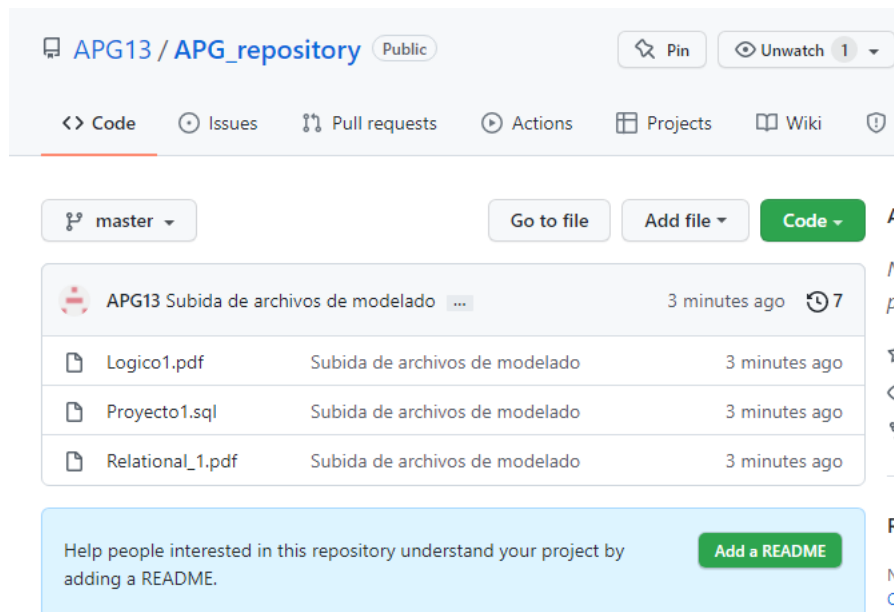
Por último, un push al repositorio.

```
APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git push origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 2 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 83.07 KiB | 27.69 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/APG13/APG_repository.git
   fbe72ae..edd9273  master -> master

APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ git status
On branch master
nothing to commit, working tree clean

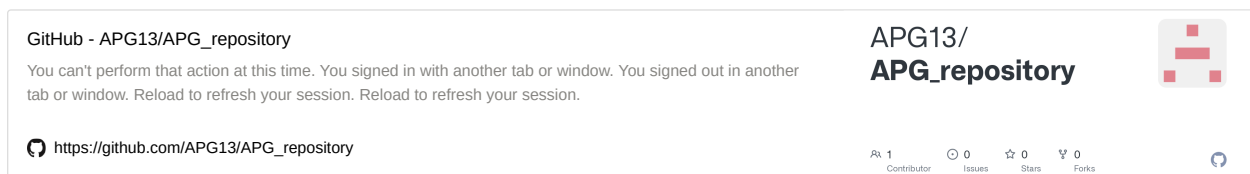
APG@DESKTOP-DEKLF18 MINGW64 /c/APG_repositorio (master)
$ |
```

Captura del repositorio una vez subidos los archivos.



## APG\_repository

A continuación dejo un enlace a mi repositorio:



FIN