

PRACTICA 01 - CARLOS JOSÉ TORRE GARCÍA

EJERCICIO 01:

1. Crear una base de datos con los datos de TiendaInformatica:

Paso 01: Crear base de datos:

```
-- Crear la base de datos
CREATE DATABASE tiendaInformatica;
GO
```

Paso 02: Usar la base de datos y crear las tablas:

```
USE tiendaInformatica;
GO
```

SQLQuery1.sql - PC0...formatica (sa (52))*

```
-- Crear tabla Fabricante
CREATE TABLE Fabricante (
    FabricanteID INT PRIMARY KEY IDENTITY(1,1),
    Nombre NVARCHAR(100) NOT NULL
);
GO

-- Crear tabla Productos
CREATE TABLE Productos (
    ProductoID INT PRIMARY KEY IDENTITY(1,1),
    NombreProducto NVARCHAR(100) NOT NULL,
    Precio DECIMAL(10, 2) NOT NULL,
    FabricanteID INT FOREIGN KEY REFERENCES Fabricante(FabricanteID)
);
GO

-- Crear tabla Clientes
CREATE TABLE Clientes (
    ClienteID INT PRIMARY KEY IDENTITY(1,1),
    NombreCliente NVARCHAR(100) NOT NULL,
    Email NVARCHAR(100)
);
GO

-- Crear tabla Pedidos
CREATE TABLE Pedidos (
    PedidoID INT PRIMARY KEY IDENTITY(1,1),
    ClienteID INT FOREIGN KEY REFERENCES Clientes(ClienteID),
    FechaPedido DATETIME DEFAULT GETDATE(),
    Total DECIMAL(10, 2)
);
GO
```

75 %

Messages

Commands completed successfully.

Completion time: 2023-06-25T11:15:00.6849862-05:00

Paso 03: Añadir los datos:

```
-- Insertar datos en la tabla Fabricante
INSERT INTO Fabricante VALUES('Asus');
INSERT INTO Fabricante VALUES('Lenovo');
INSERT INTO Fabricante VALUES('Hewlett-Packard');
INSERT INTO Fabricante VALUES('Samsung');
INSERT INTO Fabricante VALUES('Seagate');
INSERT INTO Fabricante VALUES('Crucial');
INSERT INTO Fabricante VALUES('Gigabyte');
INSERT INTO Fabricante VALUES('Huawei');
INSERT INTO Fabricante VALUES('Xiaomi');
GO

-- Insertar datos en la tabla Productos
INSERT INTO Productos VALUES('Disco duro SATA3 1TB', 86.99, 5);
INSERT INTO Productos VALUES('Memoria RAM DDR4 8GB', 120, 6);
INSERT INTO Productos VALUES('Disco SSD 1 TB', 150.99, 4);
INSERT INTO Productos VALUES('GeForce GTX 1050Ti', 185, 7);
INSERT INTO Productos VALUES('GeForce GTX 1080 Xtreme', 755, 6);
INSERT INTO Productos VALUES('Monitor 24 LED Full HD', 202, 1);
INSERT INTO Productos VALUES('Monitor 27 LED Full HD', 245.99, 1);
INSERT INTO Productos VALUES('Portátil Yoga 520', 559, 2);
INSERT INTO Productos VALUES('Portátil Ideapad 320', 444, 2);
INSERT INTO Productos VALUES('Impresora HP Deskjet 3720', 59.99, 3);
INSERT INTO Productos VALUES('Impresora HP Laserjet Pro M26nw', 180, 3);
GO
```

5 %

Messages

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

EJERCICIO 02: Crear un usuario con su contraseña

2. Crear un Login de SQL Server para el Empleado de Ventas:

```
-- Crear un login de SQL Server
CREATE LOGIN Carlospractica WITH PASSWORD = 'Josecarlos2',
CHECK_POLICY = ON;
GO
```

110 %

Messages

Commands completed successfully.

Completion time: 2025-06-25T11:22:34.8064547-05:00

EJERCICIO 03: Creamos un usuario para el login de la tiendaInformatica

3. Crear un Usuario de Base de Datos para el Login en TiendaEjemplo:

```
USE tiendaInformatica;  
GO  
  
-- Crear un usuario de base de datos para el login EmpleadoVentas  
CREATE USER Carlospractica FOR LOGIN Carlospractica;  
GO
```

EJERCICIO 04:

Creamos un rol para la base de datos y le asignamos permisos:

4. Crear un Rol de Base de Datos Personalizado para "Ventas" y Asignar Permisos:

```
USE tiendaInformatica;  
GO  
  
-- Crear un rol personalizado llamado RolVentas  
CREATE ROLE Usuariocarlos;  
GO  
  
-- Otorgar permisos al rol solo sobre la tabla Pedidos  
GRANT SELECT ON Pedidos TO Usuariocarlos;  
GRANT UPDATE ON Pedidos TO Usuariocarlos;  
GO
```

EJERCICIO 05:

Agregamos al usuario al rol

5. Agregar el Usuario EmpleadoVentas al Rol RolVentas:

```
-- Agregar el usuario Carlospractica al rol Usuariocarlos  
ALTER ROLE Usuariocarlos ADD MEMBER Carlospractica;  
GO
```

110 %



Messages

Commands completed successfully.

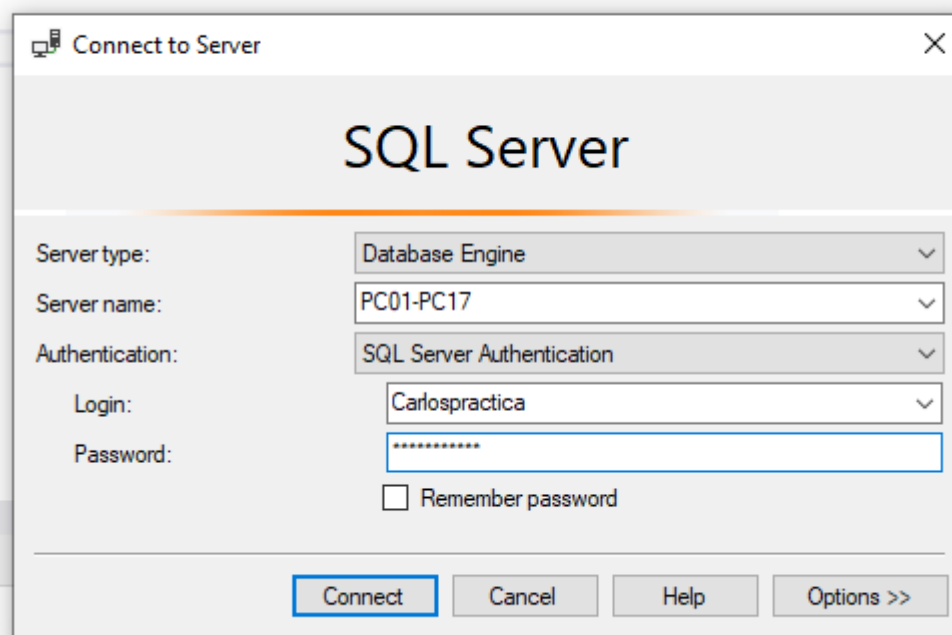
Completion time: 2025-06-25T11:34:09.5281418-05:00

EJERCICIO 06:

Prueba de seguridad

6. Probando la Configuración de Seguridad

Paso 01: Iniciamos sesion con nuestro usuario y contraseña, en este caso “Carlospractica” y la contraseña es “Josecarlos2”

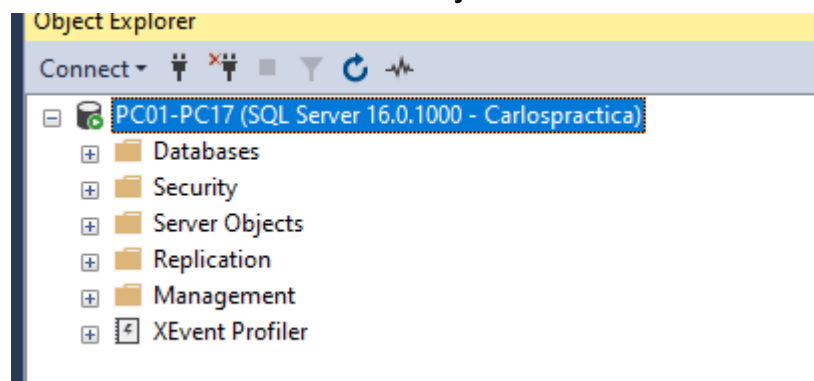


The image shows the 'Connect to Server' dialog box in SQL Server. The title bar reads 'Connect to Server'. The main heading is 'SQL Server'. The dialog contains the following fields and options:

- Server type: Database Engine (dropdown)
- Server name: PC01-PC17 (dropdown)
- Authentication: SQL Server Authentication (dropdown)
- Login: Carlospractica (dropdown)
- Password: (password field with masked characters)
- ☐ Remember password

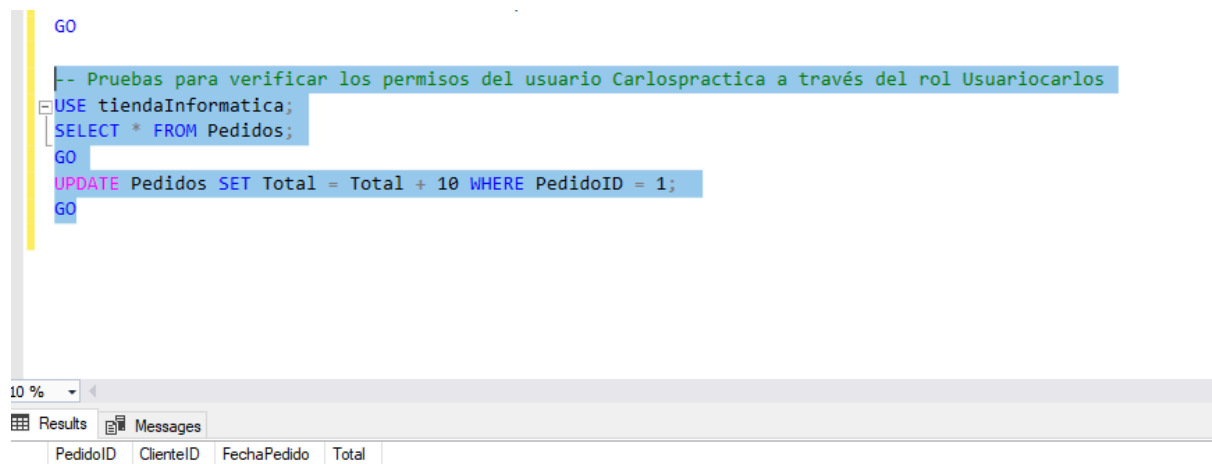
At the bottom, there are four buttons: 'Connect' (highlighted with a blue border), 'Cancel', 'Help', and 'Options >>'.

Paso 02: Nos conectamos y debería entrar:

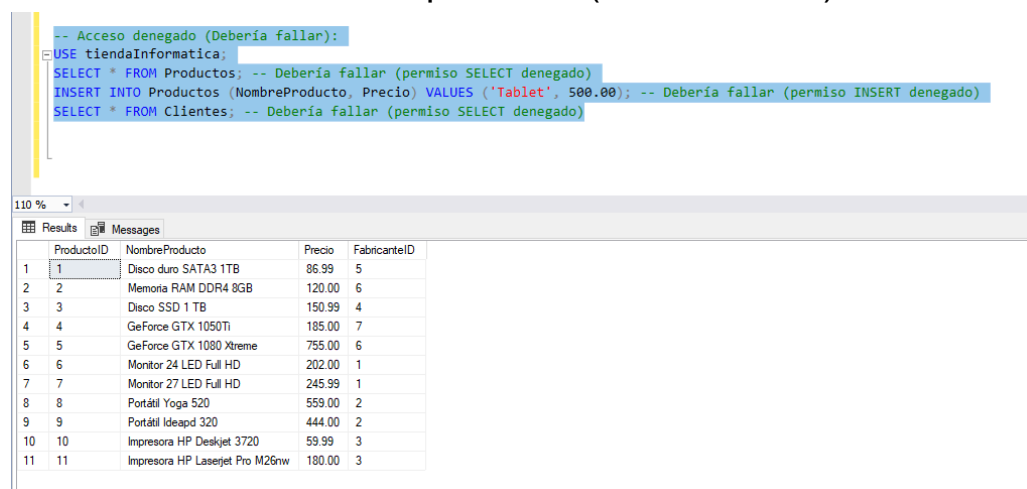


Paso 03: Verificamos los permisos (Debería funcionar)

- Acceso permitido (Debería funcionar):



Paso 04: Verificamos los permisos (Debería fallar)



EJERCICIO 07: CONFIGURACIÓN DE TDE

Configurar Cifrado Transparente de Datos (TDE):

PASO 01:

Crear una Clave Maestra de Base de Datos (Database Master Key - DMK) en la base de datos master

Protegida por una contraseña. Esta clave se usa para cifrar otras claves.

```
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'CONTRASEÑASEGURA';  
GO
```

PASO 02:

Crear un Certificado en la base de datos master

Este certificado se usará para proteger la clave de cifrado de la base de datos

```
CREATE CERTIFICATE TDE_Certificado_TiendaInformatica  
WITH SUBJECT = 'Certificado para TDE de tiendaInformatica';  
GO
```

PASO 03:

Crear la Clave de Cifrado de la Base de Datos (Database Encryption Key - DEK)

Esta clave es la que realmente cifrará los datos de la base de datos.

Está protegida por el certificado que creamos en master.

```
CREATE DATABASE ENCRYPTION KEY  
WITH ALGORITHM = AES_256  
ENCRYPTION BY SERVER CERTIFICATE TDE_Certificado_TiendaInformatica;  
GO
```

PASO 04:

Habilitar TDE en la base de datos tiendaInformatica

```
ALTER DATABASE tiendaInformatica  
SET ENCRYPTION ON;  
GO
```

PASO 05:

Verificar el estado del cifrado (puede tardar un poco en bases de datos grandes)

```
-- Verificar el estado del cifrado (puede tardar un poco en bases de datos grandes)
SELECT
    db.name,
    db.is_encrypted,
    dm.encryption_state,
    dm.percent_complete,
    dm.key_algorithm,
    dm.key_length
FROM
    sys.databases db
LEFT OUTER JOIN
    sys.dm_database_encryption_keys dm
ON
    db.database_id = dm.database_id
WHERE db.name = 'tiendaInformatica';
```

Results Messages Client Statistics

name	is_encrypted	encryption_state	percent_complete	key_algorithm	key_length
tiendaInformatica	0	NULL	NULL	NULL	NULL