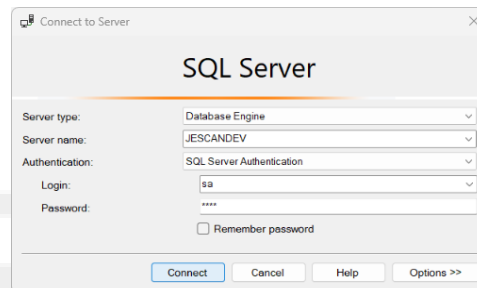


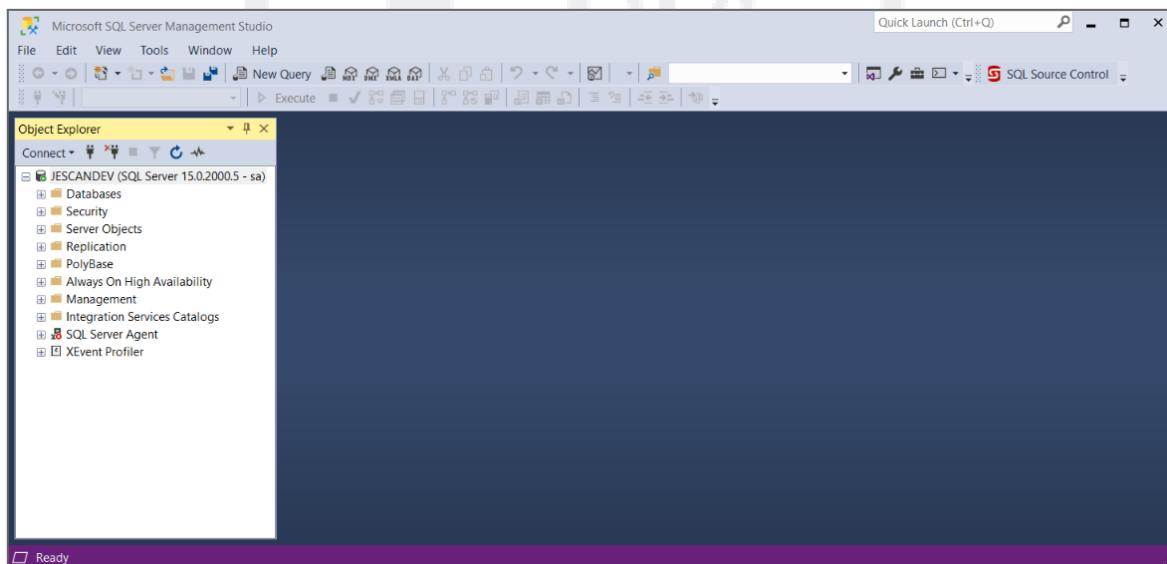
# Implementación de Sales Clothes con T-SQL

## Conexión a MS SQL Server

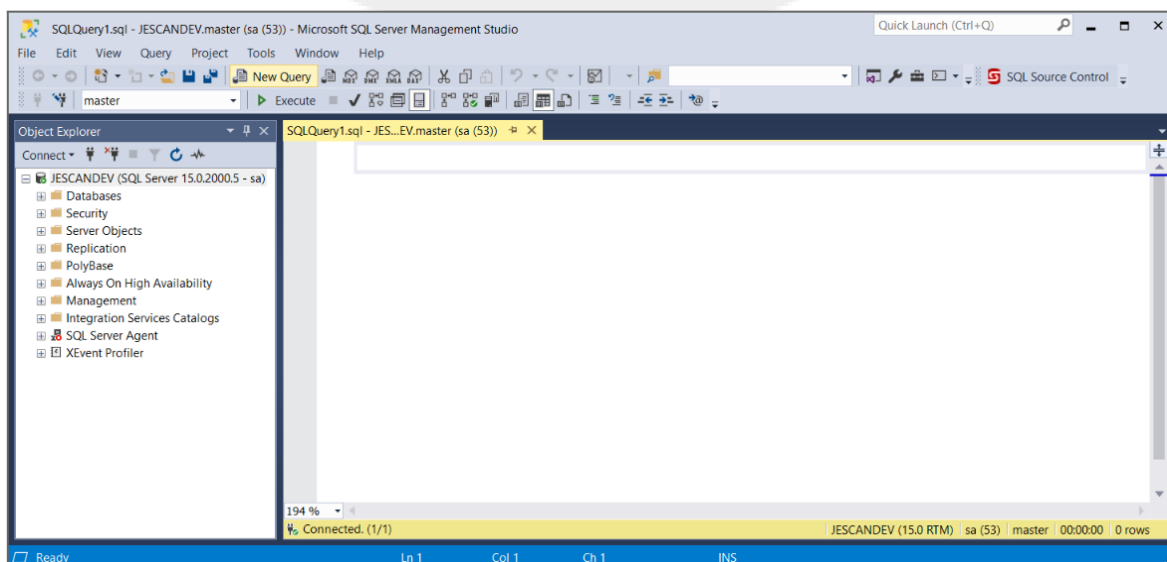
1. Ejecutar *Microsoft SQL Server Management Studio*.
2. Acceder con el usuario super administrador *sa* y su respectiva *contraseña*. Hacer click en *Connect*



3. Ahora ya estamos dentro del motor de base de datos.



4. Para empezar a escribir nuestro código SQL debemos hacer click en el botón *New Query*.



## Base de datos

### Crear base de datos

1. Procedemos a crear la base de datos **db\_SalesClothes**

```
/* Si la base de datos ya existe la eliminamos */  
DROP DATABASE IF EXISTS db_SalesClothes;
```

```
/* Crear base de datos Sales Clothes */  
CREATE DATABASE db_SalesClothes;
```

### Poner en uso

1. Ponemos en uso la base de datos utilizando código T-SQL

```
/* Poner en uso la base de datos */  
USE db_SaleClothes;
```

### Eliminar base de datos

1. Para eliminar una base de datos, esta no debe estar en uso, la base de datos en uso deberá ser master

```
/* Poner en uso base de datos master */  
USE master;
```

2. Ahora procedemos a eliminar la base de datos db\_SalesClothes

```
/* Eliminar base de datos */  
DROP DATABASE db_SalesClothes;
```

### Actividad 1

1. Utilizando código T-SQL deberá crear la base de datos db\_SalesClothes.
2. Agregar una captura de pantalla verificando la existencia de la base de datos db\_SalesClothes

## Tablas

### Crear tabla

1. Crear tabla client

```
/* Crear tabla client */
CREATE TABLE client
(
    id int,
    type_document char(3),
    number_document char(15),
    names varchar(60),
    last_name varchar(90),
    email varchar(80),
    cell_phone char(9),
    birthdate date,
    active bit
    CONSTRAINT client_pk PRIMARY KEY (id)
);
```

### Ver estructura de tabla

1. Ver estructura de tabla con código T-SQL

```
/* Ver estructura de tabla client */
EXEC sp_columns @table_name = 'client';
```

### Listar tablas

1. Para listar las tablas creadas en la base de datos db\_SalesClothes:

```
/* Listar tablas de la base de datos db_SalesClothes */
SELECT * FROM INFORMATION_SCHEMA.TABLES;
```

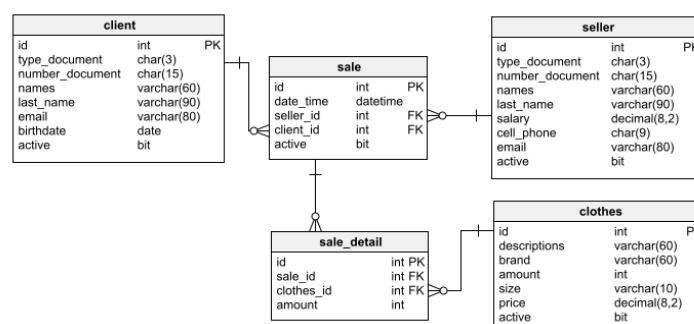
### Eliminar tabla

1. Utilizamos código T-SQL para eliminar la tabla client

```
/* Eliminar tabla client */
DROP TABLE client;
```

### Actividad 2

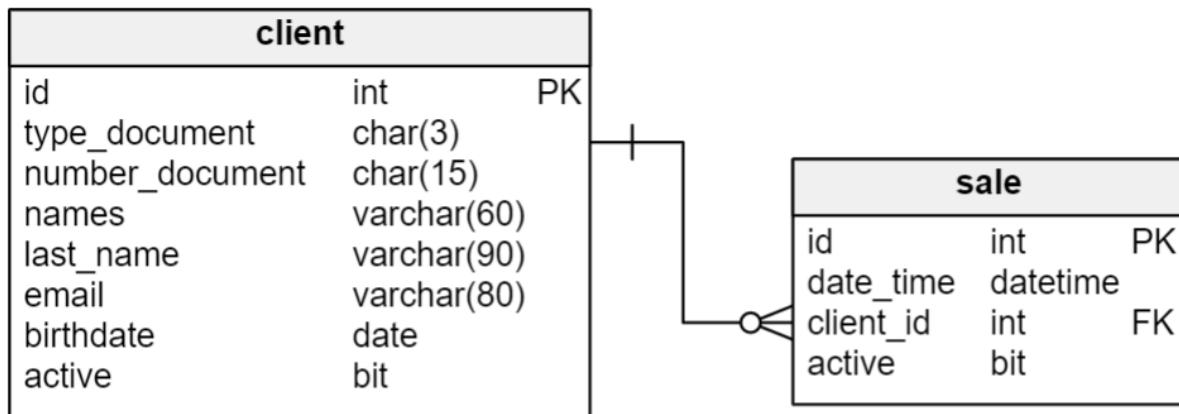
1. Utilizando código SQL crear las tablas de acuerdo a la siguiente imagen:



## Relaciones

### Crear relación

- De acuerdo a nuestro diseño físico tenemos la relación de la tabla client con la tabla sale.



- Utilizando código SQL la relación sale\_client sería:

```

/* Relacionar tabla sale con tabla client */
ALTER TABLE sale
    ADD CONSTRAINT sale_client FOREIGN KEY (client_id)
    REFERENCES client (id)
    ON UPDATE CASCADE
    ON DELETE CASCADE
GO
  
```

### Listar relaciones

- Vamos a verificar si se ha creado la relación, para ello utilizamos el siguiente código SQL:

```

/* Ver relaciones creadas entre las tablas de la base de datos */
SELECT
    fk.name [Constraint],
    OBJECT_NAME(fk.parent_object_id) [Tabla],
    COL_NAME(fc.parent_object_id,fc.parent_column_id) [Columna FK],
    OBJECT_NAME (fk.referenced_object_id) AS [Tabla base],
    COL_NAME(fc.referenced_object_id, fc.referenced_column_id) AS [Columna PK]
FROM
    sys.foreign_keys fk
    INNER JOIN sys.foreign_key_columns fc ON (fk.OBJECT_ID = fc.constraint_object_id)
GO
  
```

- Verificamos en la siguiente imagen de resultado el listado con las relaciones creadas hasta el momento.

Results Messages					
	Constraint	Tabla	Columna FK	Tabla base	Columna PK
1	sale_client	sale	client_id	client	id

## Eliminar relación

1. Vamos a eliminar la relación utilizando código T-SQL

```
/* Eliminar una relación */
ALTER TABLE sale
    DROP CONSTRAINT sale_client
GO
```

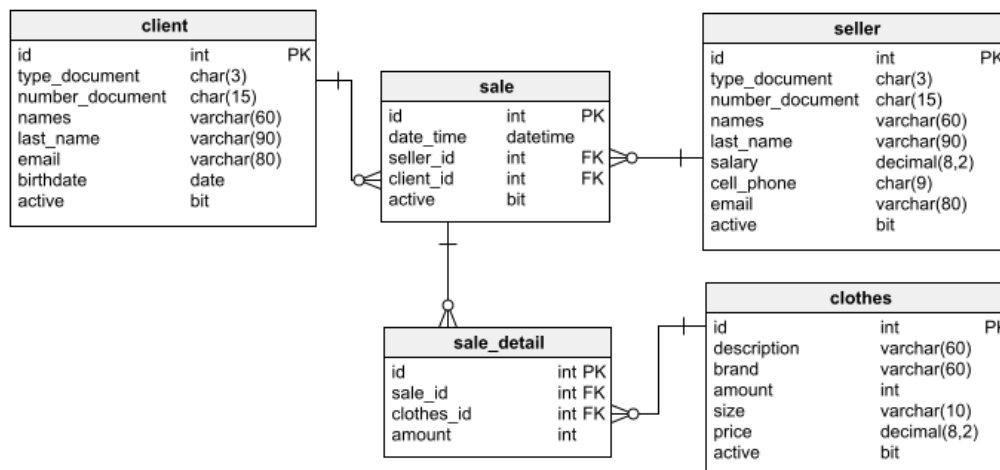
2. Vamos a verificar si se ha eliminado la relación, para ello utilizamos el siguiente código T-SQL:

```
/* Ver relaciones creadas entre las tablas de la base de datos */
SELECT
    fk.name [Constraint],
    OBJECT_NAME(fk.parent_object_id) [Tabla],
    COL_NAME(fc.parent_object_id,fc.parent_column_id) [Columna FK],
    OBJECT_NAME (fk.referenced_object_id) AS [Tabla base],
    COL_NAME(fc.referenced_object_id, fc.referenced_column_id) AS [Columna PK]
FROM
    sys.foreign_keys fk
    INNER JOIN sys.foreign_key_columns fc ON (fk.OBJECT_ID = fc.constraint_object_id)
GO
```

3. Veremos ahora que no habrá resultados reflejando así que la relación ha sido eliminada.

## Actividad 3

1. Utilizando código T-SQL crear las relaciones entre las tablas de acuerdo a la siguiente imagen:



2. Utilizando T-SQL deberá listar las relaciones creadas cuyo resultado deberá ser:

	Constraint	Tabla	Columna FK	Tabla base	Columna PK
1	sale_client	sale	client_id	client	id
2	sale_seller	sale	seller_id	seller	id
3	sale_detail_clothes	sale_detail	clothes_id	clothes	id
4	sale_detail_sale	sale_detail	sale_id	sale	id

3. Eliminar base de datos db\_SalesClothes

## Implementar script funcional

1. Vamos a proceder a crear nuestro script T-SQL funcional de implementación de base de datos, para ello deberá hacerlo en el siguiente orden:
  - ◆ Crear base de datos **db\_SalesClothes**.
  - ◆ Poner en uso la base de datos **db\_SalesClothes**.
  - ◆ Crear tablas maestras: (client, seller, clothes).
  - ◆ Crear tablas transaccionales: (sale y sale\_detail).
  - ◆ Crear relaciones entre tablas.
2. El funcionamiento de este script debe ser verificado al momento de ejecutar todas las líneas de código T-SQL la base de datos db\_SalesClothes debe ser implementada sin ningún problema.
3. Crear un repositorio en Github y versionar correctamente el script de la base de datos **db\_SalesClothes**.