

# CONVERSIÓN DE TABLAS A BD EN WORKBENCH

## ESCUELA:

```
1 • CREATE DATABASE escuela_modificada;
2
```

1-Creamos la base de datos

```
1 • CREATE DATABASE escuela_m
2 • USE escuela_modificada;
```

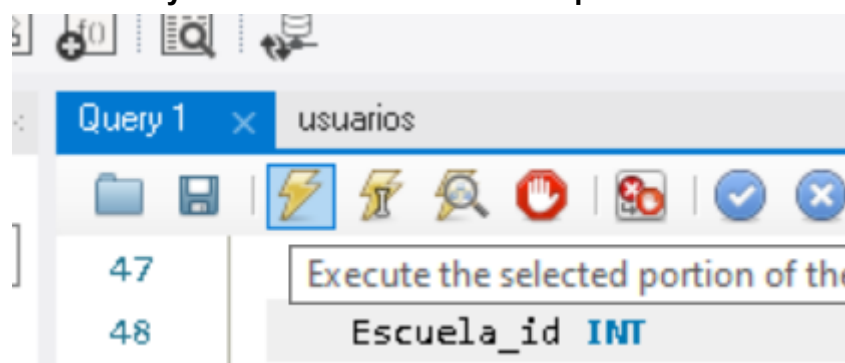
2-Utilizamos USE para continuar con la base de datos deseada

```
3
4 • CREATE TABLE ingresos (
5     idIngreso INT PRIMARY KEY,
6     Asunto VARCHAR(255),
7     Cantidad DECIMAL
```

3-Continuamos creando las tablas en base las tablas de nuestro archivo de powerpoint

```
3
4 • CREATE TABLE ingresos (
5     idIngreso INT PRIMARY KEY,
6     Asunto VARCHAR(255),
7     Cantidad DECIMAL(10,2),
8     Id_Escuela INT,
9     Producto VARCHAR(100),
10    Escuela_id_escuela INT
11 );
12
13 • CREATE TABLE cafeteria (
14     idProducto VARCHAR(100) PRIMARY KEY,
15     Precio DECIMAL(8,2),
16     Escuela_id INT
17 );
```

4-Utilizamos decimal ya que con este podemos llevar el control de los dígitos enteros antes de la coma en mi caso 10 y los números decimales después de la coma en mi caso 2.



5-Ejecutamos el codigo y verificamos que este no de error.



up

loc

▼

Fi

X

Connection Options

Connect to DBMS

Select Schemas

Retrieve Objects

Select Objects

Reverse Engineer

Results

Set Parameters for Connecting to a DBMS

Stored Connection: 

LocalHost

▼

Select from saved connection settings

Connection Method: 

Standard (TCP/IP)

▼

Method to use to connect to the RDBMS

Parameters

SSL

Advanced

Hostname: 

127.0.0.1

 Port: 

3306

Name or IP address of the server host - and TCP/IP port.

Username: 

root

Name of the user to connect with.

Password: 

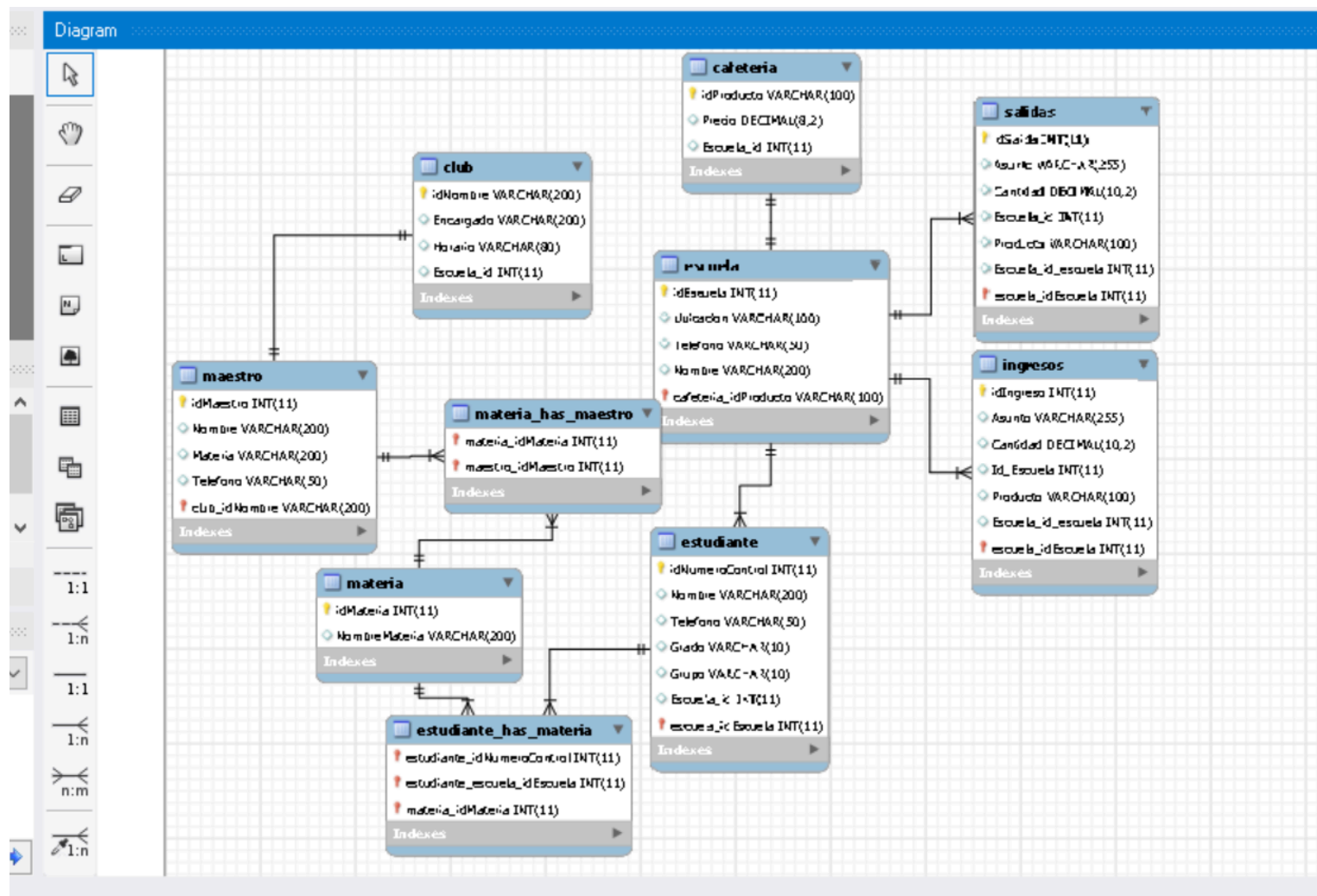
Store in Vault ...

Clear

The user's password. Will be requested later if it's not set.

- ☒ escuela\_modificada
- ☐ phpmyadmin
  - ☐ sakila
  - ☐ sakila2
  - ☐ test

## 7-Continuamos con las relaciones y la organización del esquema.



8-Organizamos las tablas y creamos las relaciones con las opciones que se nos dan a la izquierda.

## xbox:

1-Repetimos y creamos la base de datos y las tablas desde mysql en el query1, ahora con las tablas de excel de videojuegos y redes sociales en el caso del tema que escogio cada quien.



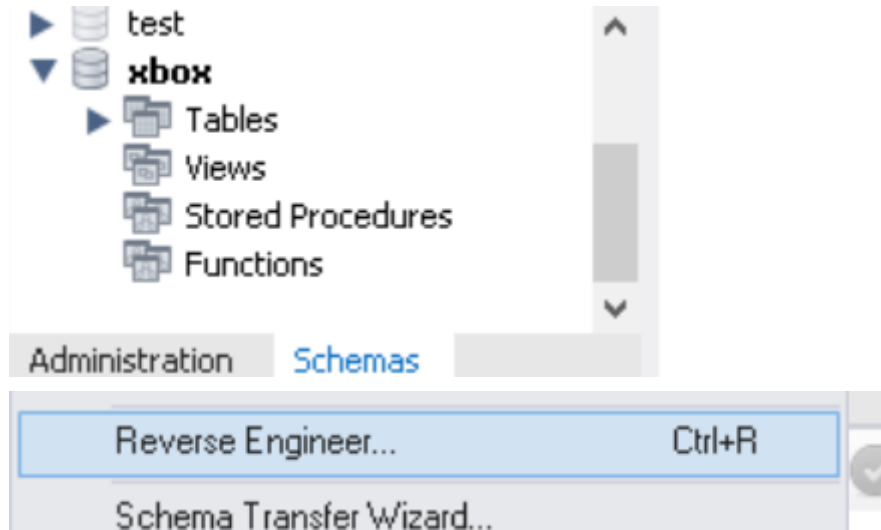
```
1 • CREATE DATABASE xbox;
2 • USE xbox;
3
4 • CREATE TABLE tienda (
5     videojuego VARCHAR(100),
6     categoria VARCHAR(100),
7     precio DECIMAL(8,2),
8     PRIMARY KEY (videojuego)
9 );
```

	STORE		INGRESO		
	videojuego		jugador		
	categoria		compra		
	precio		fecha		
			total		
				VIDEOJUEGO	
JUGADOR				nombre	
gamertag				categoria	
correo				descripcion	
contraseña				precio	
amigos				desarrollador	
metodo de pago				version	
biblioteca					
fecha					
puntos					
	CATEGORIA				
	nombre de categoria				
	descripcion				

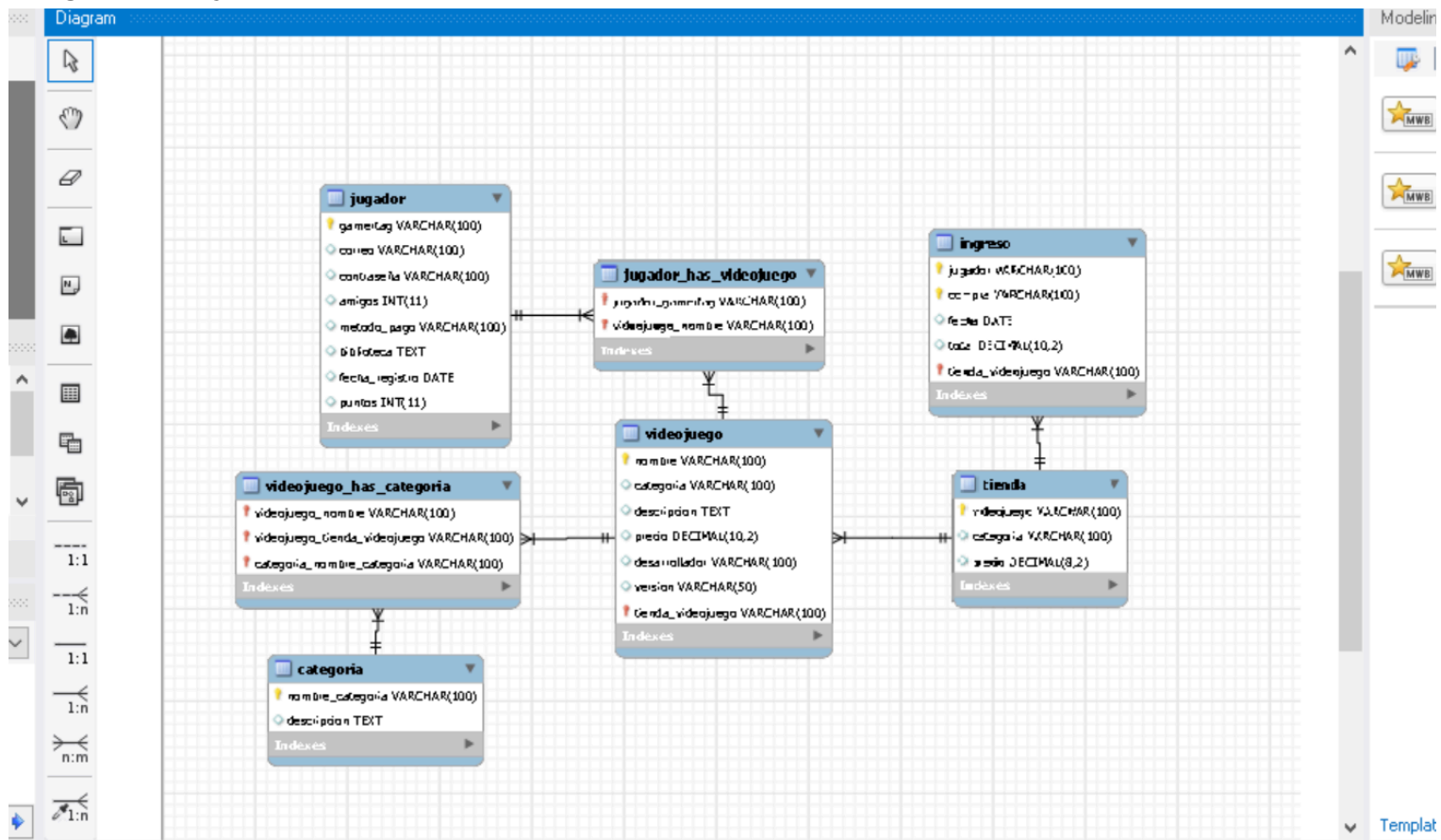
2-En cada tabla requerimos colocar una llave primaria para un control mayor de los registros.

```
• CREATE TABLE videojuego (
    nombre VARCHAR(100) PRIMARY KEY,
    categoria VARCHAR(100),
    descripcion TEXT,
    precio DECIMAL(10,2),
    desarrollador VARCHAR(100),
    version VARCHAR(50)
);
```

3-Una vez finalizada la última tabla verificamos y nos dirigimos para dar click en reverse engineer



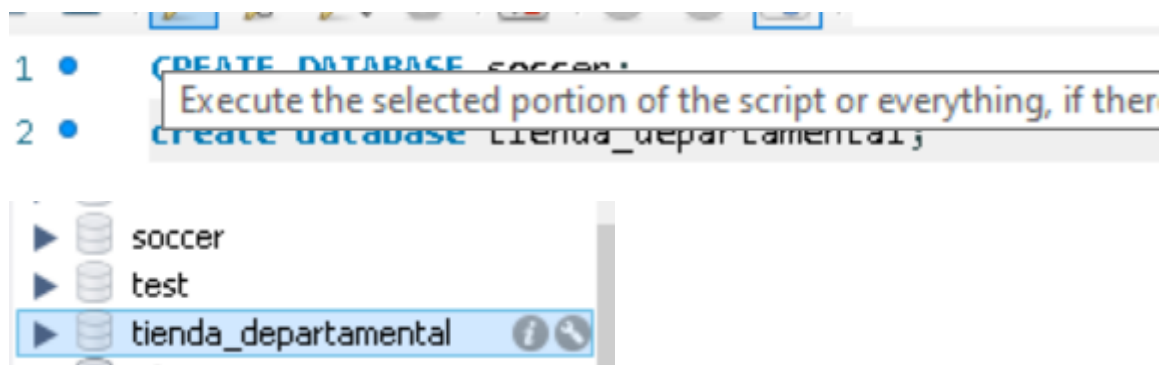
4-Organizamos y colocamos las relaciones entre las tabla de la base de datos.



# TIENDA DEPARTAMENTAL Y SOCCER:

1-Creamos las bases de datos de las dos.





2-Una vez creadas comenzamos con la creación de las tablas basándose en las de EXCEL que ya tenemos creadas.

es recomendable para telefonos usar varchar o int?

Para almacenar números de teléfono, es más recomendable usar `VARCHAR` en lugar de `INT`. Aquí están algunas razones:

1. **Formatos Variables:** Los números de teléfono a menudo contienen caracteres adicionales como paréntesis, guiones, o espacios. Por ejemplo: `(555) 123-4567` o `+52 1 800 123 4567`. Usar `VARCHAR` te permite almacenar estos caracteres.
2. **Longitud Variable:** Los números de teléfono pueden variar en longitud dependiendo del país y la región. Usar `VARCHAR` te proporciona la flexibilidad necesaria.
3. **No Realizan Cálculos:** No necesitas realizar cálculos con los números de teléfono. Usar `INT` no ofrece ningún beneficio en este caso.

Por lo tanto, es mejor usar `VARCHAR` para almacenar números de teléfono, por ejemplo:

Sql

Copiar

```
telefono VARCHAR(15)
```

Esto te dará la flexibilidad para incluir cualquier formato necesario. ¡Espero que esto aclare tus dudas!



Enviar un mensaje a Copilot

Think Deeper



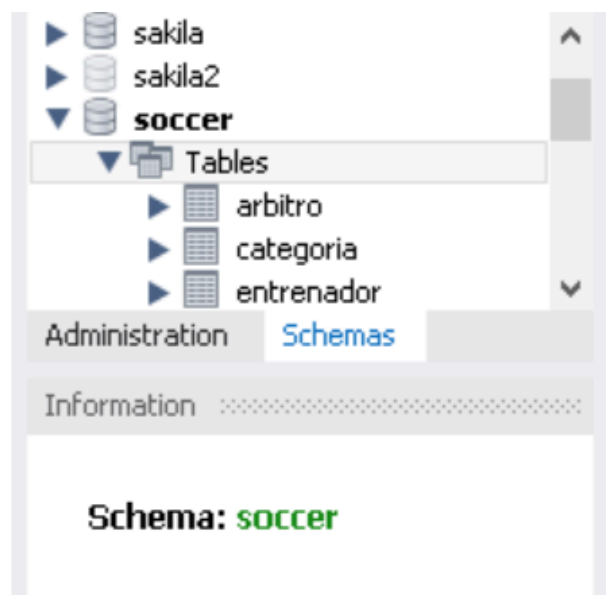
3-En los telefonos usamos varchar ya que es mas recomendable para los numeros de telefonos como se dice arriba.

```
1 ● CREATE DATABASE soccer;
2 ● create database tienda_departamental;
3
4 ● use soccer;
5
6 ● ⊖ CREATE TABLE categoria (
7     nombre VARCHAR(100) PRIMARY KEY,
8     descripcion TEXT,
9     equipo VARCHAR(100)
10 ) ;
11
12 ● ⊖ CREATE TABLE jugador (
13     numero_jugador INT PRIMARY KEY,
14     nombre VARCHAR(100),
15     categoria VARCHAR(100),
16     telefono VARCHAR(15)
17 ) ;
```

NOTA:USAMOS LLAVES FORANEAS PARA RELACIONAR DE MANERA MAS PRECISA LA BD

```
Query 1
PRIMARY KEY (asunto, fecha)
);
CREATE TABLE arbitro (
    nombre VARCHAR(100) PRIMARY KEY,
    telefono VARCHAR(15),
    direccion TEXT
);
CREATE TABLE partido (
    encuentro VARCHAR(100),
    fecha DATE,
    ubicacion VARCHAR(100),
    resultado VARCHAR(100),
    equipo_local VARCHAR(100),
    equipo_visitante VARCHAR(100),
    PRIMARY KEY (encuentro, fecha),
    FOREIGN KEY (equipo_local) REFERENCES equipo(nombre,
    FOREIGN KEY (equipo_visitante)
```

4-Creamos las tablas y colocamos la llave primaria en todas para llevar un mejor control y manejo.



```
2
3 ● ○ CREATE TABLE categoria (
4     nombre VARCHAR(100) PRIMARY KEY,
5     descripcion TEXT,
6     equipo VARCHAR(100)
7     );
8
9 ● ○ CREATE TABLE jugador (
10     numero_jugador INT PRIMARY KEY,
11     nombre VARCHAR(100),
12     categoria VARCHAR(100)
```

5-Verificamos que todas se crearon correctamente y continuamos en reverse engineer para crear el modelo entidad relacion de esta base de datos.

## Select Schemas to Reverse Engineer



Select the schemas you want to include:

- ☐ escuela\_modificada
- ☐ phpmyadmin
- ☐ sakila
- ☐ sakila2
- ☒ soccer
- ☐ test
- ☐ tienda\_departamental
- ☐ xbox

Back


Next

Cancel

**CORREGI TODO YA QUE LAS RELACIONES NO LAS CREE DESDE EL MISMO SQL, SI NO CON EL MODELO ENTIDAD RELACIÓN...**

```
42         asunto VARCHAR(255),
43         cantidad DECIMAL(10,2),
44         fecha DATE,
45         PRIMARY KEY (asunto, fecha)
46     );
47
48 • CREATE TABLE arbitro (
49     nombre VARCHAR(100) PRIMARY KEY,
50     telefono VARCHAR(15),
51     direccion TEXT,
52     FOREIGN KEY (partido) REFERENCES equipo(nombre)
53 );
54
55 • CREATE TABLE partido (
56     encuentro VARCHAR(100),
57     fecha DATE,
```

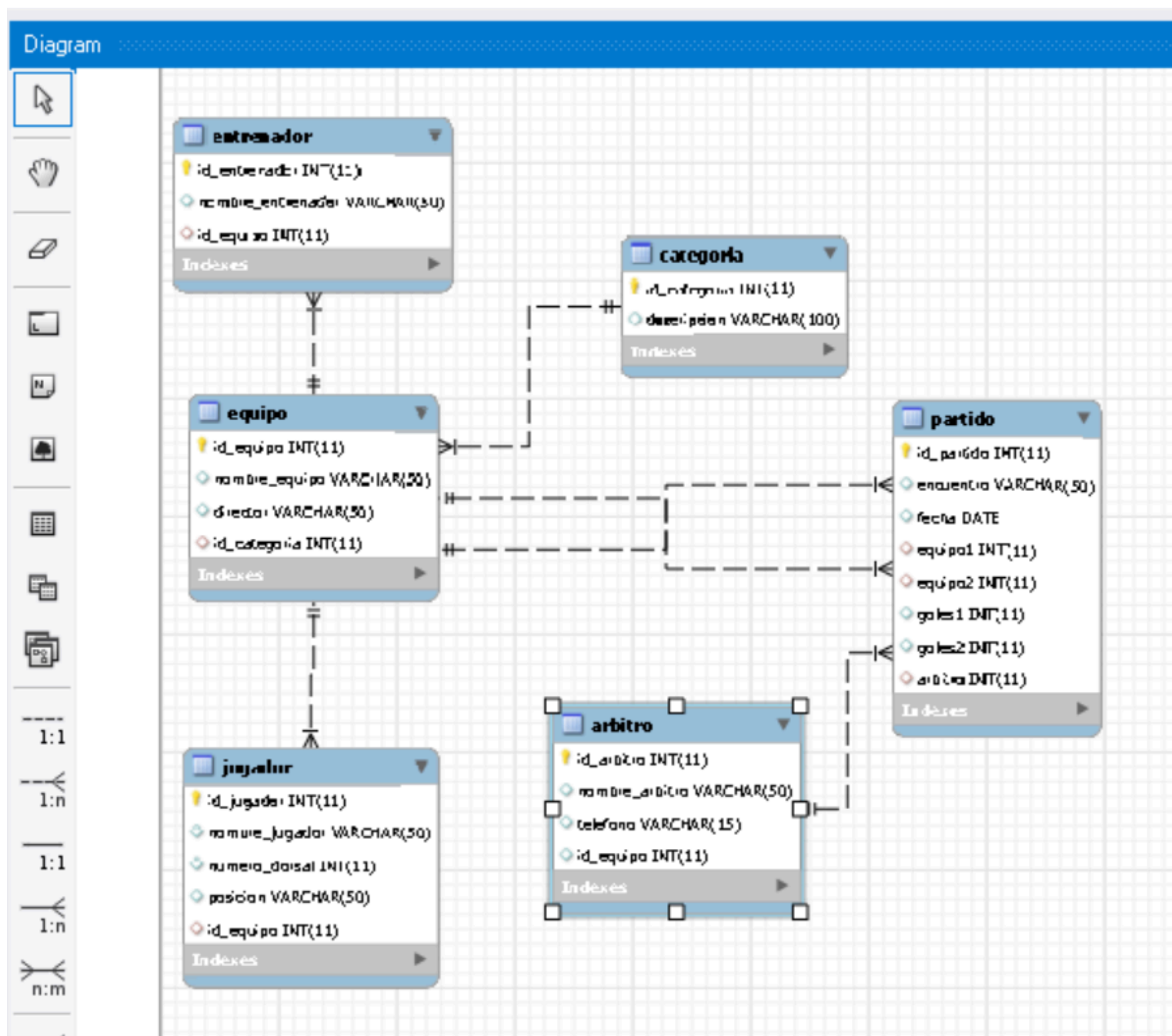
**1- DAMOS DROP EN TODAS LAS BASES DE DATOS PARA EN BASE AL CODIGO ANTERIORMENTE USADO CREAR LLAVES FORANEAS.**



Limit to 1000 rows

```
7      fecha DATE,  
8      equipo1 INT,  
9      equipo2 INT,  
0      goles1 INT,  
1      goles2 INT,  
2      arbitro INT,  
3      FOREIGN KEY (equipo1) REFERENCES equipo(id_equipo),  
4      FOREIGN KEY (equipo2) REFERENCES equipo(id_equipo),  
5      FOREIGN KEY (arbitro) REFERENCES arbitro(id_arbitro)  
6  );
```

```
7  
8  ● ○ CREATE TABLE entrenador(  
9      id_entrenador INT PRIMARY KEY AUTO_INCREMENT,  
0      nombre_entrenador VARCHAR(50),  
1      id_equipo INT,  
2      FOREIGN KEY (id_equipo) REFERENCES equipo(id_equipo)  
3  );  
4  
5
```



AHORA CON XBOX...

1-Creamos la base de datos y relacionamos las tablas con llaves foraneas para la plataforma de xbox.



Limit to 1000 rows

```

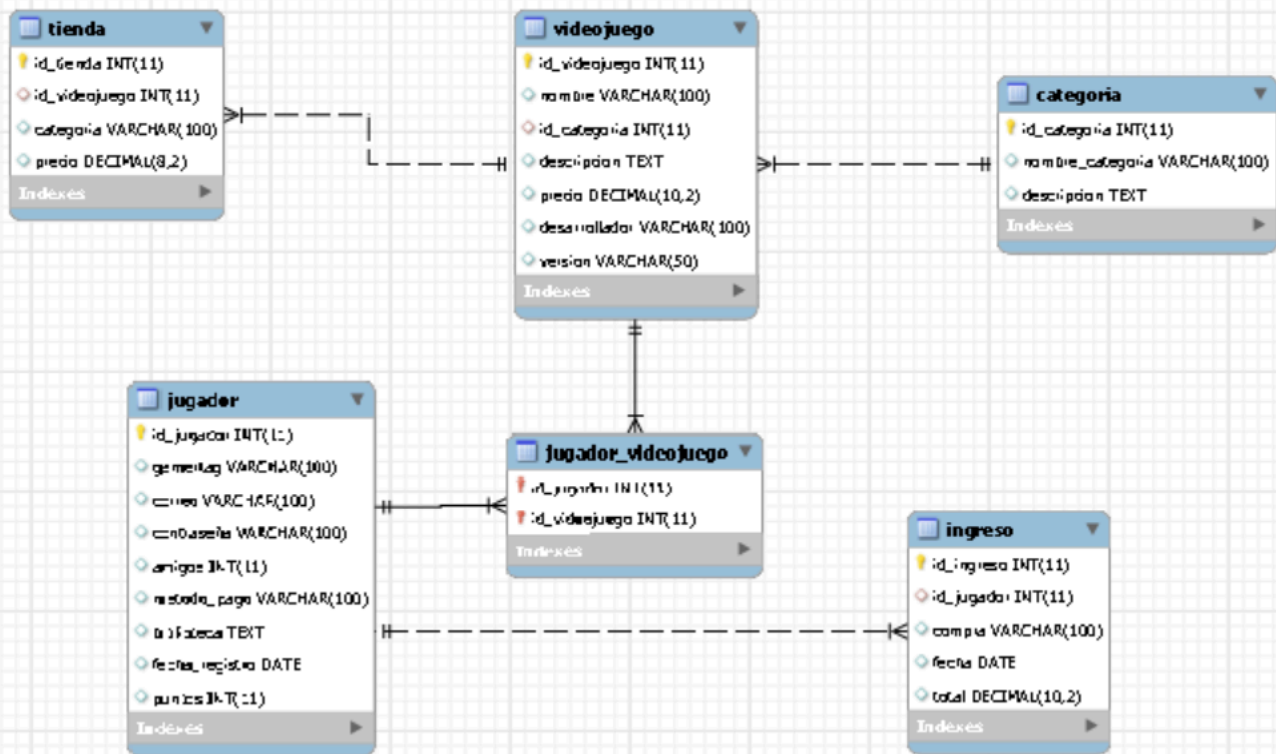
1 • CREATE DATABASE xbox;
2
3 • USE xbox;
4
5 • CREATE TABLE categoria (
6     id_categoria INT PRIMARY KEY AUTO_INCREMENT,
7     nombre_categoria VARCHAR(100),
8     descripcion TEXT
9 );
10
11 • CREATE TABLE videojuego (
12     id_videojuego INT PRIMARY KEY AUTO_INCREMENT,
13     nombre VARCHAR(100),
14     id_categoria INT,
15     descripcion TEXT,
16     precio DECIMAL(10,2),
17     desarrollador VARCHAR(100),
18     version VARCHAR(50),
19     FOREIGN KEY (id_categoria) REFERENCES categoria(id_categoria)

```

Action Output				
#	Time	Action	Message	Duration / Fetch
✓ 1	08:00:44	CREATE DATABASE escuela_modificada	1 row(s) affected	0.000 sec
✓ 2	08:00:44	USE escuela_modificada	0 row(s) affected	0.000 sec
✓ 3	08:00:44	CREATE TABLE ingresos ( idIngreso INT PRIMARY KEY, Asunto VARCHAR...	0 row(s) affected	0.015 sec
✓ 4	08:00:44	CREATE TABLE cafeteria ( idProducto VARCHAR(100) PRIMARY KEY, Pre...	0 row(s) affected	0.016 sec
✓ 5	08:00:44	CREATE TABLE salidas ( idSalida INT PRIMARY KEY, Asunto VARCHAR(2...	0 row(s) affected	0.000 sec
✓ 6	08:00:44	CREATE TABLE Escuela ( idEscuela INT PRIMARY KEY, Ubicacion VARC...	0 row(s) affected	0.015 sec
✓ 7	08:00:44	CREATE TABLE estudiante ( idNumeroControl INT PRIMARY KEY, Nombre ...	0 row(s) affected	0.000 sec
✓ 8	08:00:44	CREATE TABLE club ( idNombre VARCHAR(200) PRIMARY KEY, Encargad...	0 row(s) affected	0.016 sec
✓ 9	08:00:44	CREATE TABLE maestro ( idMaestro INT PRIMARY KEY, Nombre VARCHA...	0 row(s) affected	0.000 sec
✓ 10	08:00:44	CREATE TABLE materia ( idMateria INT PRIMARY KEY, NombreMateria VA...	0 row(s) affected	0.016 sec
✓ 11	08:01:14	SELECT * FROM escuela_modificada.cafeteria LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓ 12	08:01:16	SELECT * FROM escuela_modificada.club LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓ 13	08:01:17	SELECT * FROM escuela_modificada.escuela LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓ 14	08:01:18	SELECT * FROM escuela_modificada.estudiante LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓ 15	08:01:19	SELECT * FROM escuela_modificada.ingresos LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓ 16	08:15:52	CREATE DATABASE xbox	1 row(s) affected	0.000 sec
✓ 17	08:15:52	USE xbox	0 row(s) affected	0.000 sec
✓ 18	08:15:52	CREATE TABLE tienda ( videojuego VARCHAR(100), categoria VARCHAR(...	0 row(s) affected	0.015 sec
✓ 19	08:15:52	CREATE TABLE ingreso ( jugador VARCHAR(100), compra VARCHAR(100)...	0 row(s) affected	0.000 sec
✓ 20	08:15:52	CREATE TABLE jugador ( gamertag VARCHAR(100) PRIMARY KEY, correo ...	0 row(s) affected	0.016 sec
✓ 21	08:15:52	CREATE TABLE videojuego ( idVideojuego INT PRIMARY KEY AUTO_INCREMENT, ...	0 row(s) affected	0.016 sec

```
1
2
3 • USE xbox;
4
5 • CREATE TABLE jugador_videojuego (
6     id_jugador INT,
7     id_videojuego INT,
8     FOREIGN KEY (id_jugador) REFERENCES jugador(id_jugador),
9     FOREIGN KEY (id_videojuego) REFERENCES videojuego(id_videojuego),
10    PRIMARY KEY (id_jugador, id_videojuego)
11 );
```

me falto modificar para que el jugador pueda tener varios juegos y un juego pueda ser comprado o jugado por varios jugadores.



AHORA LA ESCUELA Y LA TIENDA DEPARTAMENTAL TAMBIEN CORREGIREMOS:

```
1
2 • CREATE DATABASE Escuela;
3 • USE Escuela;
4
5 • CREATE TABLE maestro (
6     id INT PRIMARY KEY AUTO_INCREMENT,
7     nombre VARCHAR(100),
8     materia VARCHAR(100),
9     telefono VARCHAR(15)
10 );
11
12 • CREATE TABLE grupo (
13     id INT PRIMARY KEY AUTO_INCREMENT,
14     grado INT,
15     grupo VARCHAR(50),
16     ubicacion VARCHAR(100),
17     maestro_id INT,
18     FOREIGN KEY (maestro_id) REFERENCES maestro(id)
19 );
--
```

Query 1 x SQL File 4

Limit to 1000 rows

```

21 • CREATE TABLE estudiantes (
22     id INT PRIMARY KEY AUTO_INCREMENT,
23     numero_de_control VARCHAR(50),
24     nombre VARCHAR(100),
25     grado INT,
26     grupo_id INT,
27     FOREIGN KEY (grupo_id) REFERENCES grupo(id)
28 );
29
30 • CREATE TABLE materia (
31     id INT PRIMARY KEY AUTO_INCREMENT,
32     nombre VARCHAR(100),
33     maestro_id INT,
34     horario VARCHAR(50),
35     FOREIGN KEY (maestro_id) REFERENCES maestro(id)
36 );
37
38 • CREATE TABLE CLUBES (
39     id INT PRIMARY KEY AUTO_INCREMENT,
40     nombre VARCHAR(100),

```

```

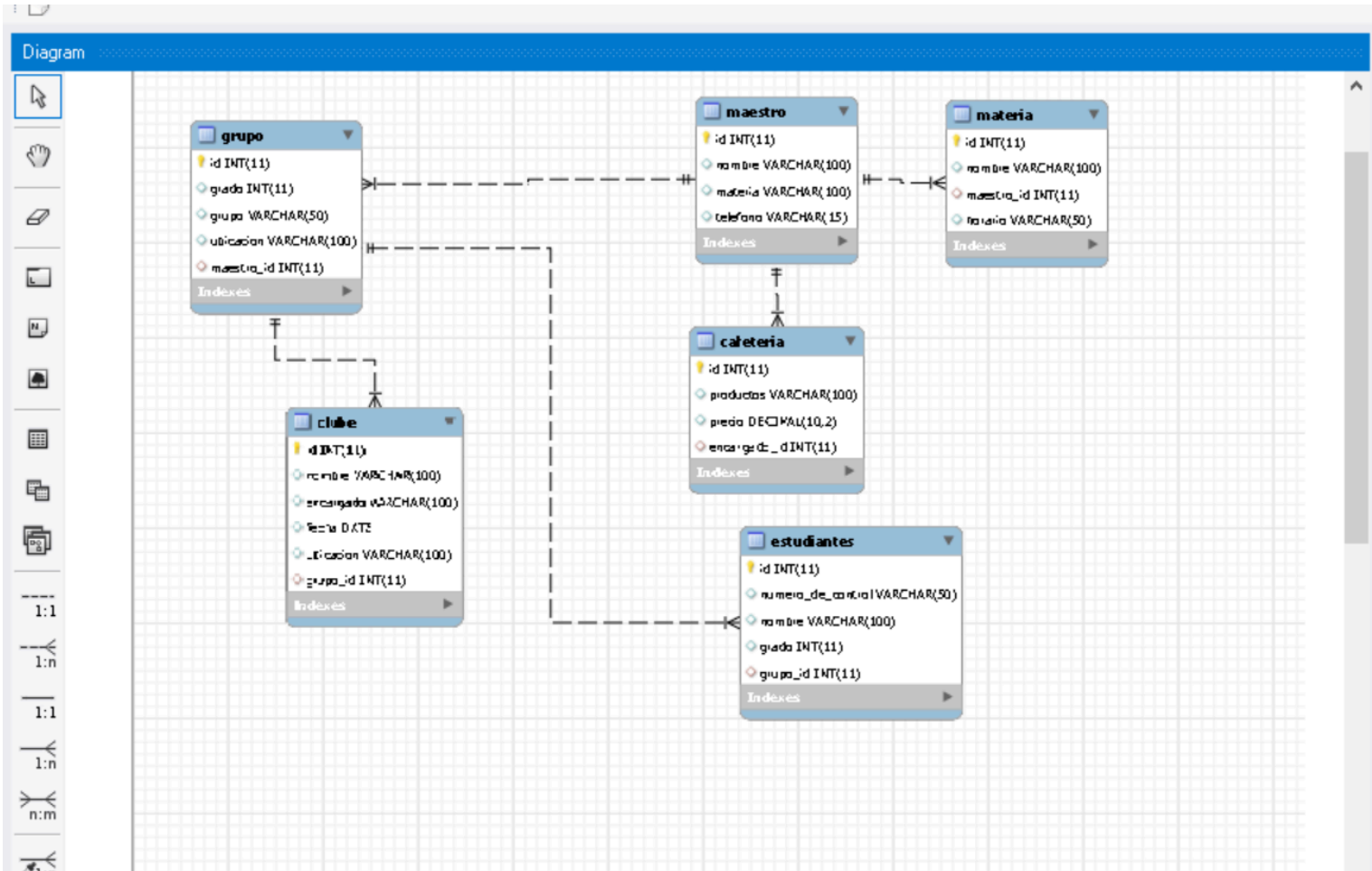
38 • CREATE TABLE CLUBE (
39     id INT PRIMARY KEY AUTO_INCREMENT,
40     nombre VARCHAR(100),
41     encargado VARCHAR(100),
42     fecha DATE,
43     ubicacion VARCHAR(100),
44     grupo_id INT,
45     FOREIGN KEY (grupo_id) REFERENCES grupo(id)
46 );
47
48 • CREATE TABLE CAFETERIA (
49     id INT PRIMARY KEY AUTO_INCREMENT,
50     productos VARCHAR(100),
51     precio DECIMAL(10, 2),
52     encargado_id INT,
53     FOREIGN KEY (encargado_id) REFERENCES maestro(id)
54 );
55
56

```

Output

Action Output

1-Ahora lo convertimos en modelo entidad relacion para verificar que todo este correcto.



TIENDA DEPARTAMENTAL----->

```

1 • CREATE DATABASE tiendadepartamental;
2 • USE tiendadepartamental;
3
4
5 • CREATE TABLE departamentos (
6     clave_departamento INT PRIMARY KEY AUTO_INCREMENT,
7     seccion VARCHAR(100),
8     encargado VARCHAR(100)
9 );
10
11 • CREATE TABLE marca (
12     clave_marca INT PRIMARY KEY AUTO_INCREMENT,
13     nombre VARCHAR(100),
14     descripcion TEXT
15 );
16
17 • CREATE TABLE cliente (
18     clave_cliente INT PRIMARY KEY AUTO_INCREMENT,
19     nombre VARCHAR(100)

```

MySQL Model\*

Database Server Tools Scripting Help

Query 1 x SQL File 4\*

```

41     clave_articulo INT PRIMARY KEY AUTO_INCREMENT,
42     nombre VARCHAR(100),
43     descripcion TEXT,
44     stock INT,
45     marca_id INT,
46     departamento_id INT,
47     FOREIGN KEY (marca_id) REFERENCES marca(clave_marca),
48     FOREIGN KEY (departamento_id) REFERENCES departamentos(clave_departamento)
49 );
50
51 • CREATE TABLE venta_articulo (
52     id INT PRIMARY KEY AUTO_INCREMENT,
53     venta_id INT,
54     articulo_id INT,
55     FOREIGN KEY (venta_id) REFERENCES ventas(clave_venta),
56     FOREIGN KEY (articulo_id) REFERENCES articulo(clave_articulo)
57 );
58
59

```



