

Nombre de la práctica	Diagrama de entidad relación		No.	
Asignatura: Taller de bases de datos	Carrera :	Ing eniería de sistemas computacionales	Duración de la práctica (Hrs)	2hrs

I. Competencia(s) específica(s):

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro): Aula

III. Material empleado:

Computadora y gestor de base de datos

IV. Desarrollo de la práctica:

Problema 1:

```
qll80-localhost  ✘ <mysql> G_TIENDA_CLIENTE
-- CREAMOS LA BASE DE DATOS DE GESTION_TIENDA_CLIENTES (PROBLEMA 1)
CREATE DATABASE G_TIENDA_CLIENTE;
USE G_TIENDA_CLIENTE;
```

```
•-- CREAMOS LA TABLA DE TIENDA
CREATE TABLE Tienda (
    id_tienda INT PRIMARY KEY,
    nombre_tienda VARCHAR(100) NOT NULL,
    municipio VARCHAR(100) NOT NULL,
    calle VARCHAR(100) NOT NULL,
    colonia VARCHAR(100) NOT NULL,
    CP CHAR(5) NOT NULL
);

•-- CREAMOS LA TABLA DE CLIENTE
CREATE TABLE Cliente (
    id_cliente INT PRIMARY KEY,
    nombre_cliente VARCHAR(100) NOT NULL,
    apellido1 VARCHAR(100) NOT NULL,
    apellido2 VARCHAR(100)
);

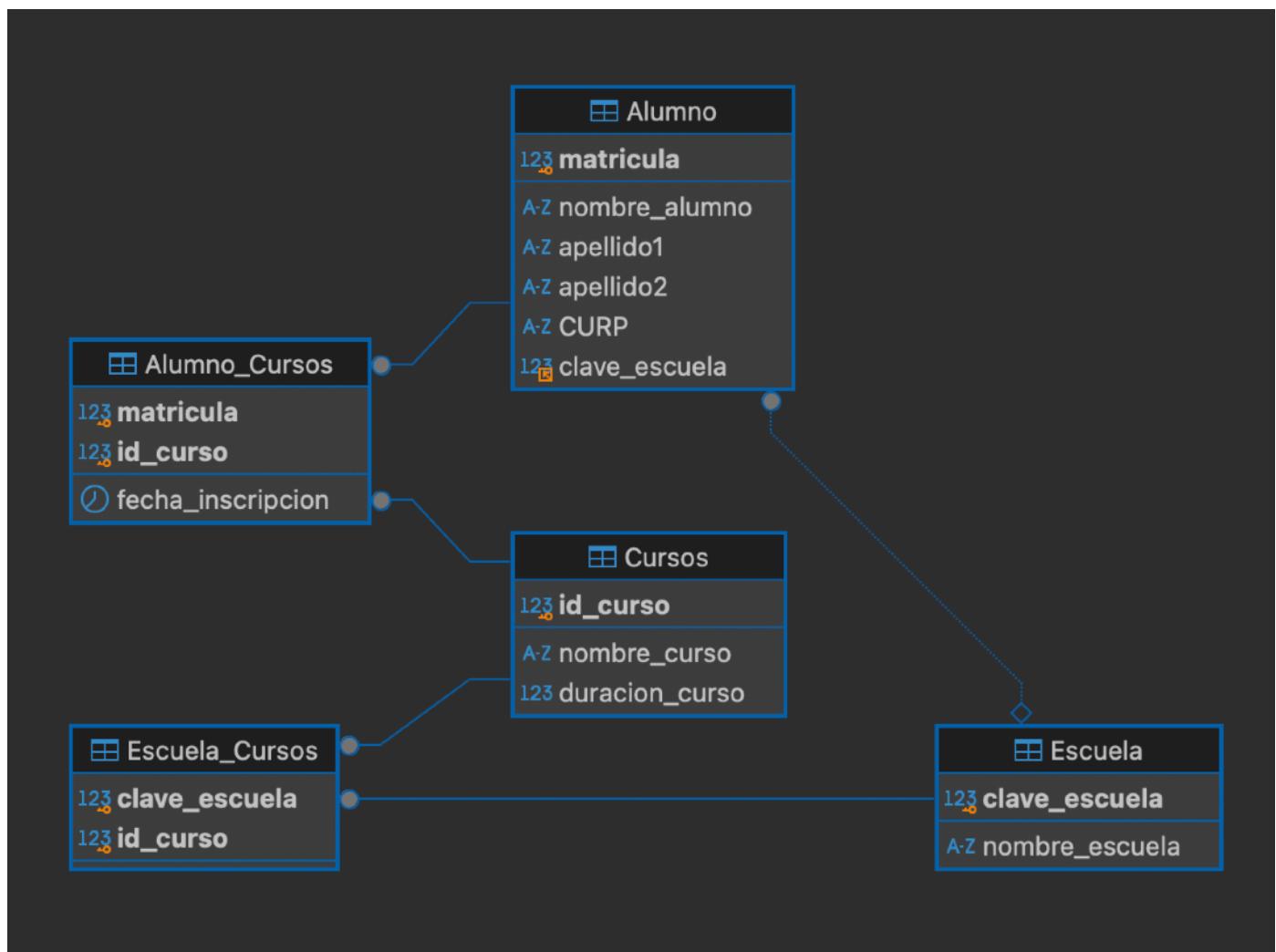
•-- CREAMOS LA TABLA DE LA RELACION DE TIENDA A CLIENTE
CREATE TABLE Tienda_Cliente (
    id_tienda INT,
    id_cliente INT,
    PRIMARY KEY (id_tienda, id_cliente),
    FOREIGN KEY (id_tienda) REFERENCES Tienda(id_tienda),
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)
);

•-- CREAMOS LA TABLA DE PRODUCTO
CREATE TABLE Producto (
    clave_producto INT PRIMARY KEY,
    nombre_producto VARCHAR(100) NOT NULL,
    precio_producto DECIMAL(10, 2) NOT NULL
);

•-- CREAMOS LA TABLA DE COMPRA
CREATE TABLE Compra (
    clave_producto INT,
    id_cliente INT,
    fecha_compra DATE NOT NULL,
    PRIMARY KEY (clave_producto, id_cliente),
    FOREIGN KEY (clave_producto) REFERENCES Producto(clave_producto),
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)
);
```

```
● -- CREAMOS LA TABLA DE TIENDA A PRODUCTO
CREATE TABLE Tienda_Producto (
    id_tienda INT,
    clave_producto INT,
    PRIMARY KEY (id_tienda, clave_producto),
    FOREIGN KEY (id_tienda) REFERENCES Tienda(id_tienda),
    FOREIGN KEY (clave_producto) REFERENCES Producto(clave_producto)
);
```

Visualización de los modelos relacionales:



Problema 2:

```
-- CREAMOS LA BASE DE DATOS DE GESTION_TIENDA_CLIENTES (PROBLEMA 1)

CREATE DATABASE G_TIENDA_CLIENTE;

USE G_TIENDA_CLIENTE;

-- CREAMOS LA TABLA DE TIENDA
CREATE TABLE Tienda (
    id_tienda INT PRIMARY KEY,
    nombre_tienda VARCHAR(100) NOT NULL,
    municipio VARCHAR(100) NOT NULL,
    calle VARCHAR(100) NOT NULL,
    colonia VARCHAR(100) NOT NULL,
    CP CHAR(5) NOT NULL
);

-- CREAMOS LA TABLA DE CLIENTE
CREATE TABLE Cliente (
    id_cliente INT PRIMARY KEY,
    nombre_cliente VARCHAR(100) NOT NULL,
    apellido1 VARCHAR(100) NOT NULL,
    apellido2 VARCHAR(100)
);

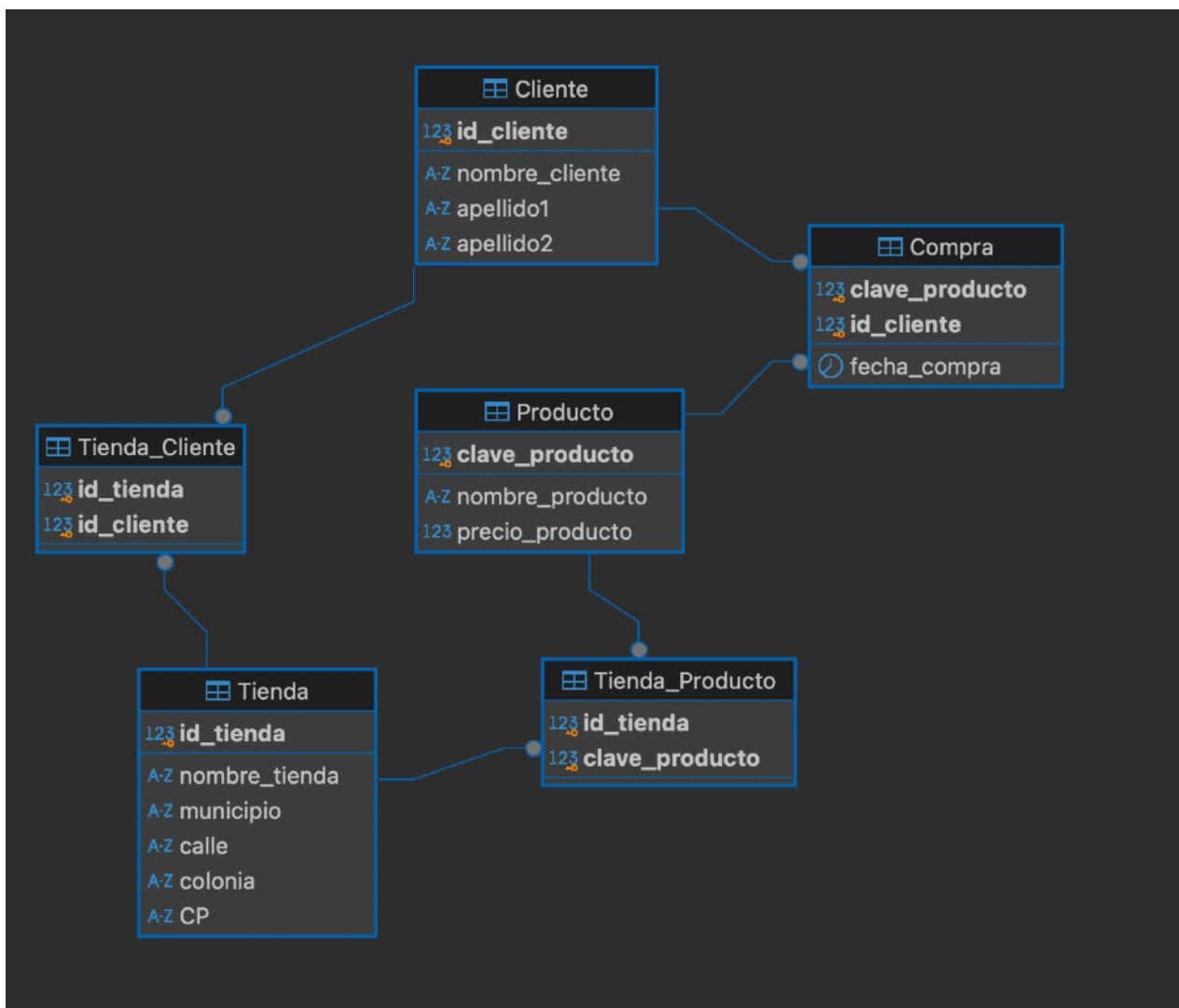
-- CREAMOS LA TABLA DE LA RELACION DE TIENDA A CLIENTE
CREATE TABLE Tienda_Cliente (
    id_tienda INT,
    id_cliente INT,
    PRIMARY KEY (id_tienda, id_cliente),
    FOREIGN KEY (id_tienda) REFERENCES Tienda(id_tienda),
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)
);

-- CREAMOS LA TABLA DE PRODUCTO
CREATE TABLE Producto (
    clave_producto INT PRIMARY KEY,
    nombre_producto VARCHAR(100) NOT NULL,
    precio_producto DECIMAL(10, 2) NOT NULL
);

-- CREAMOS LA TABLA DE COMPRA
CREATE TABLE Compra (
    clave_producto INT,
    id_cliente INT,
    fecha_compra DATE NOT NULL,
    PRIMARY KEY (clave_producto, id_cliente),
    FOREIGN KEY (clave_producto) REFERENCES Producto(clave_producto),
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)
);
```

```
● -- CREAMOS LA TABLA DE TIENDA A PRODUCTO
CREATE TABLE Tienda_Producto (
    id_tienda INT,
    clave_producto INT,
    PRIMARY KEY (id_tienda, clave_producto),
    FOREIGN KEY (id_tienda) REFERENCES Tienda(id_tienda),
    FOREIGN KEY (clave_producto) REFERENCES Producto(clave_producto)
);
```

Revisión de el modelo relacional:



Problema 3:

```
① -- CREAMOS LA BASE DE DATOS PARA LA GESTION DE PROYECTOS EN EMPRESA (PROBLEMA 3)
CREATE DATABASE G_PRO_EMP;
USE G_PRO_EMP;

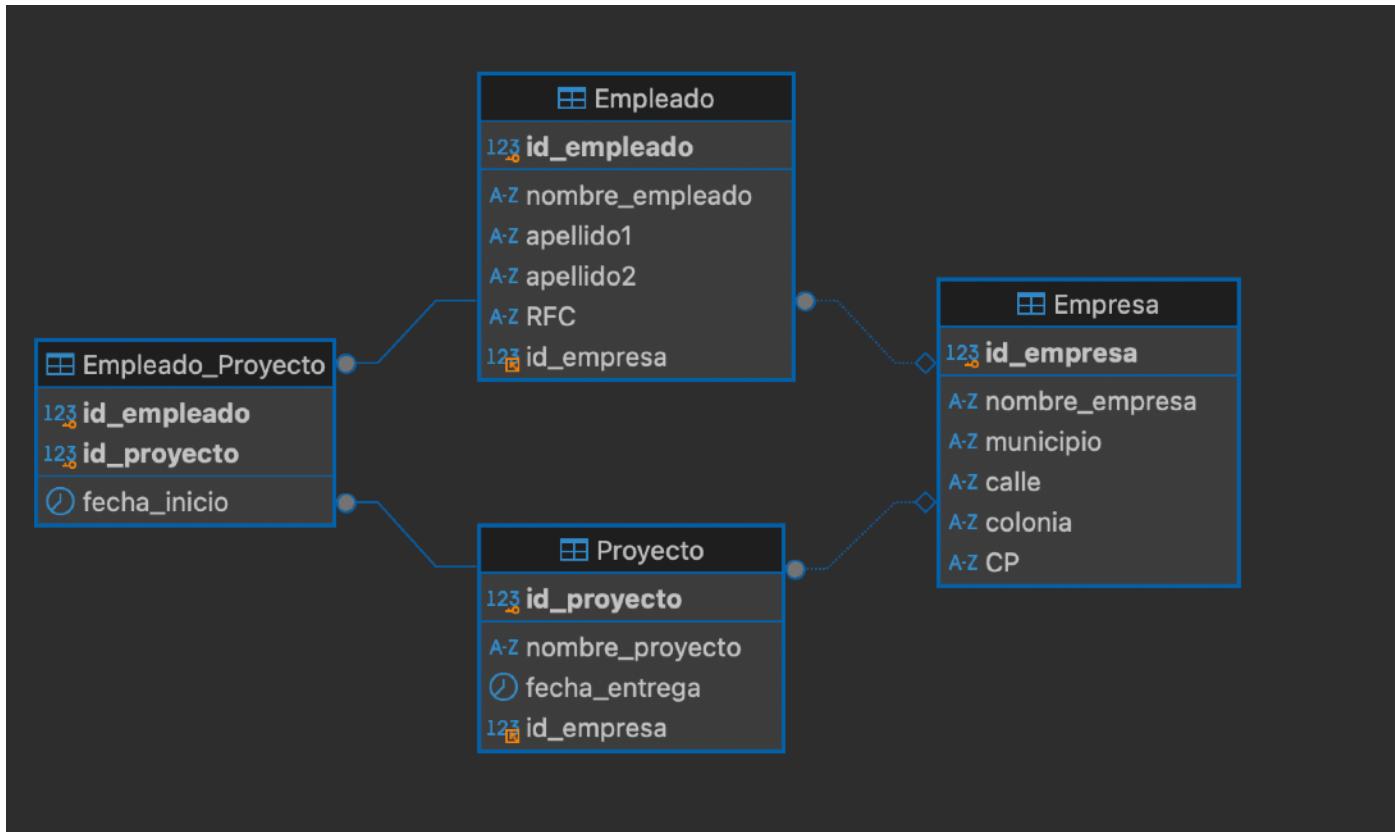
② -- CREAMOS LA TABLA DE EMPRESA
CREATE TABLE Empresa (
    id_empresa INT PRIMARY KEY,
    nombre_empresa VARCHAR(100) NOT NULL,
    municipio VARCHAR(100) NOT NULL,
    calle VARCHAR(100) NOT NULL,
    colonia VARCHAR(100) NOT NULL,
    CP CHAR(5) NOT NULL
);

③ -- CREAMOS LA TABLA DE PROYECTO
CREATE TABLE Proyecto (
    id_proyecto INT PRIMARY KEY,
    nombre_proyecto VARCHAR(100) NOT NULL,
    fecha_entrega DATE NOT NULL,
    id_empresa INT,
    FOREIGN KEY (id_empresa) REFERENCES Empresa(id_empresa)
);

④ -- CREAMOS LA TABLA DE EMPLEADO
CREATE TABLE Empleado (
    id_empleado INT PRIMARY KEY,
    nombre_empleado VARCHAR(100) NOT NULL,
    apellido1 VARCHAR(100) NOT NULL,
    apellido2 VARCHAR(100),
    RFC CHAR(13) NOT NULL,
    id_empresa INT,
    FOREIGN KEY (id_empresa) REFERENCES Empresa(id_empresa)
);

⑤ -- CREAMOS LA TABLA DE LA RELACION DE EMPLEADO A PROYECTO
CREATE TABLE Empleado_Proyecto (
    id_empleado INT,
    id_proyecto INT,
    fecha_inicio DATE NOT NULL,
    PRIMARY KEY (id_empleado, id_proyecto),
    FOREIGN KEY (id_empleado) REFERENCES Empleado(id_empleado),
    FOREIGN KEY (id_proyecto) REFERENCES Proyecto(id_proyecto)
);
```

Revisión de el modelo relacional:



Problema 4:

```
•-- CREAMOS LA BASE DE DATOS PARA LA GESTION DE CONSULTA EN HOSPITAL (PROBLEMA 4)
CREATE DATABASE G_CONSULTAS_HOS;
USE G_CONSULTAS_HOS;

•-- CREAMOS LA TABLA DE HOSPITAL
CREATE TABLE Hospital (
    id_hospital INT PRIMARY KEY,
    nombre_hospital VARCHAR(100) NOT NULL,
    municipio VARCHAR(100) NOT NULL,
    calle VARCHAR(100) NOT NULL,
    colonia VARCHAR(100) NOT NULL,
    CP CHAR(5) NOT NULL
);

•-- CREAMOS LA TABLA DE PACIENTE
CREATE TABLE Paciente (
    id_paciente INT PRIMARY KEY,
    nombre_paciente VARCHAR(100) NOT NULL,
    apellido1 VARCHAR(100) NOT NULL,
    apellido2 VARCHAR(100),
    fecha_nacimiento DATE NOT NULL,
    altura DECIMAL(5,2),
    grupo_sanguíneo VARCHAR(3),
    peso DECIMAL(5,2)
);

•-- CREAMOS LA TABLA DE LA RELACION DE HOSPITAL A PACIENTE
CREATE TABLE Hospital_Paciente (
    id_hospital INT,
    id_paciente INT,
    PRIMARY KEY (id_hospital, id_paciente),
    FOREIGN KEY (id_hospital) REFERENCES Hospital(id_hospital),
    FOREIGN KEY (id_paciente) REFERENCES Paciente(id_paciente)
);

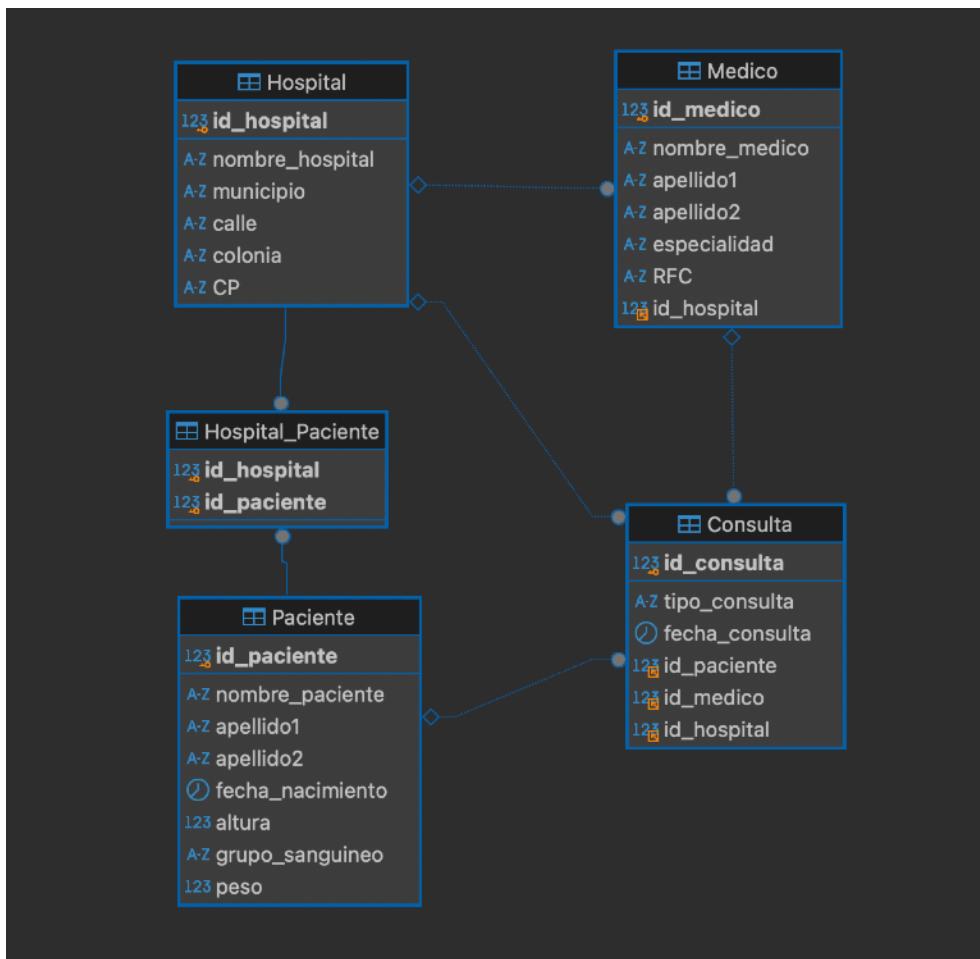
•-- CREAMOS LA TABLA DE MEDICO
CREATE TABLE Medico (
    id_medico INT PRIMARY KEY,
    nombre_medico VARCHAR(100) NOT NULL,
    apellido1 VARCHAR(100) NOT NULL,
    apellido2 VARCHAR(100),
    especialidad VARCHAR(100),
    RFC CHAR(13),
    id_hospital INT,
    FOREIGN KEY (id_hospital) REFERENCES Hospital(id_hospital)
);
```



● -- CREAMOS LA TABLA DE CONSULTA

```
CREATE TABLE Consulta (
    id_consulta INT PRIMARY KEY,
    tipo_consulta VARCHAR(100) NOT NULL,
    fecha_consulta DATE NOT NULL,
    id_paciente INT,
    id_medico INT,
    id_hospital INT,
    FOREIGN KEY (id_paciente) REFERENCES Paciente(id_paciente),
    FOREIGN KEY (id_medico) REFERENCES Medico(id_medico),
    FOREIGN KEY (id_hospital) REFERENCES Hospital(id_hospital)
);
```

Revisión de modelo relacional:



V.

Conclusiones:

Aquí es donde puse a prueba si realmente sabia o no hacer una base de datos se podría decir desde cero y me di cuenta que aunque por mas chico que sea el problema se puede complicar un poco y mas si uno la extiende o cosas así por el estilo, yo en mis bases de datos trate de poner solo atributos mas referentes a lo que lleva cada entidad y no llenarle con atributos no tan referentes, aquí también puede ser complicado el escribir o identificar que tipo de dato tiene que ser cada atributo pero creo que teniendo una idea clara, todo se hace mas fácil ya que fluye mas rápido las cosas y quedan bien