

Nombre de la práctica	Modificación del esquema de la bases de datos			No.	2
Asignatura:	Taller de Bases de Datos	Carrera:	Ingeniería en Sistemas Computacionales	Duración de la práctica (Hrs)	2

I. Nombre del alumno: Francisco David Colin Lira

II. Competencia(s) específica(s):

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

III. Material empleado: Laptop, gestor de bases de datos

IV. Desarrollo de la práctica:

PROBLEMA 1: Diseñar y modificar un esquema para una base de datos de biblioteca.

Objetivo: Diseñar en un esquema de la base de datos para gestionar la información de una biblioteca. El sistema debe ser capaz de almacenar datos sobre libros, los clientes y los préstamos realizados.

```
CREATE DATABASE BIBLIOTECA;

USE BIBLIOTECA;

CREATE TABLE Libro (
    ID_Libro INT PRIMARY KEY,
    Titulo VARCHAR(255),
    Año_Publicacion INT,
    ISBN VARCHAR(20)
);

-- Cree otra tabla para la gestion de los autores de los libros:
-- Ya que algunos autores pueden escribir varios libros
CREATE TABLE Autor (
    ID_Autor INT PRIMARY KEY,
    Nombre_autor VARCHAR(100) NOT NULL,
    Apellido1_autor VARCHAR(100) NOT NULL,
    Apellido2_autor VARCHAR(100),
);
```

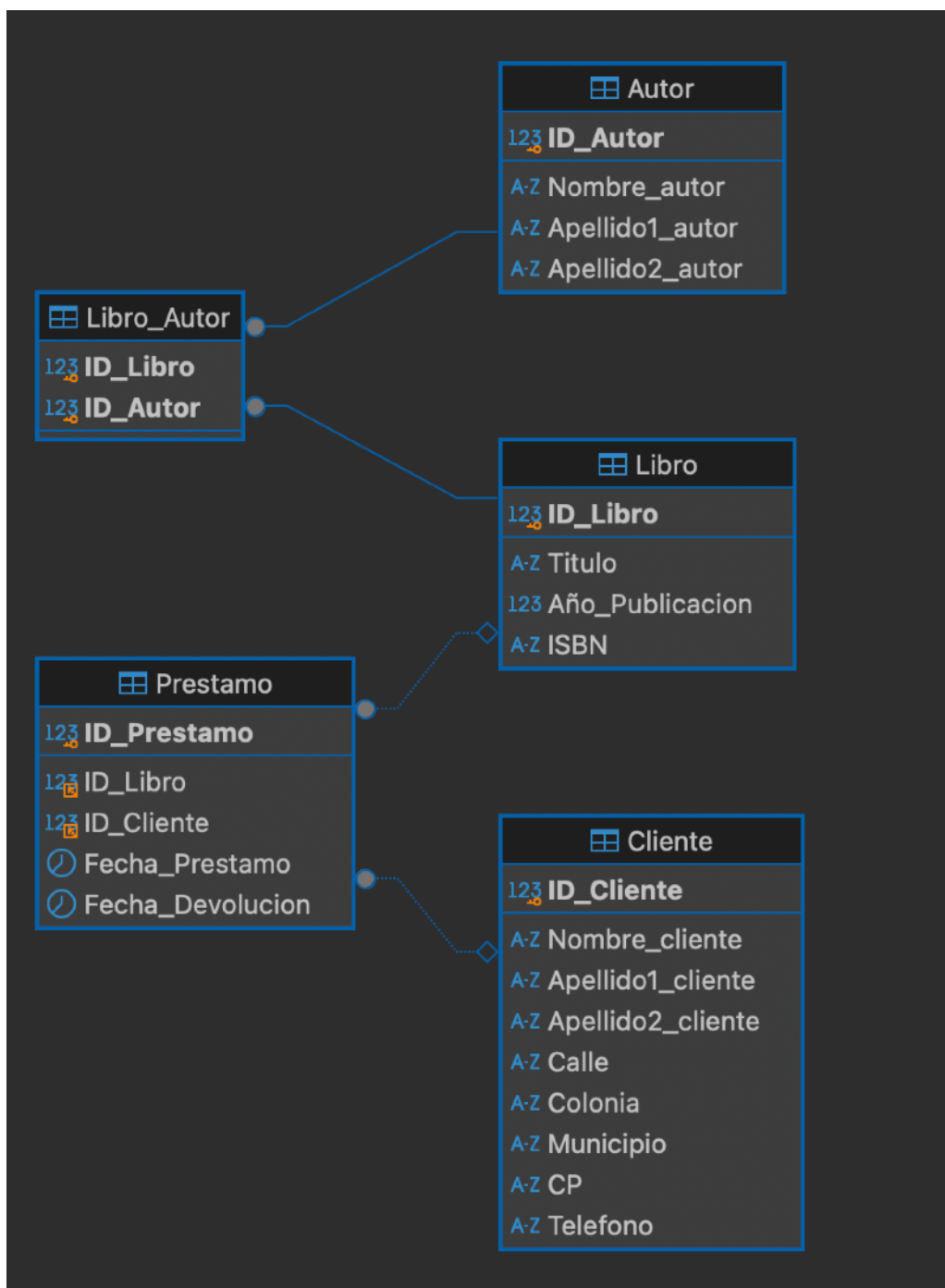


```
/*  
Como la relacion general entre Libro y Autor es N:M  
se crea una nueva tabla con la relacion de ambos  
*/  
CREATE TABLE Libro_Autor(  
    ID_Libro INT,  
    ID_Autor INT,  
    PRIMARY KEY (ID_Libro, ID_Autor),  
    FOREIGN KEY (ID_Libro) REFERENCES Libro(ID_Libro),  
    FOREIGN KEY (ID_Autor) REFERENCES Autor(ID_Autor)  
);
```

```
-- Tambien aqui se normalizo los campos de nombre y direccion del cliente  
CREATE TABLE Cliente (  
    ID_Cliente INT PRIMARY KEY,  
    Nombre_cliente VARCHAR(100) NOT NULL,  
    Apellido1_cliente VARCHAR(100) NOT NULL,  
    Apellido2_cliente VARCHAR(100),  
    Calle VARCHAR(100),  
    Colonia VARCHAR(100),  
    Municipio VARCHAR(100),  
    CP CHAR(5),  
    Telefono VARCHAR(20)  
);
```

```
CREATE TABLE Prestamo (  
    ID_Prestamo INT PRIMARY KEY,  
    ID_Libro INT,  
    ID_Cliente INT,  
    Fecha_Prestamo DATE,  
    Fecha_Devolucion DATE,  
    FOREIGN KEY (ID_Libro) REFERENCES Libro(ID_Libro),  
    FOREIGN KEY (ID_Cliente) REFERENCES Cliente(ID_Cliente)  
);
```

Así queda el diagrama:



PROBLEMA 2: Diseñar y modificar un esquema para un sistema de ventas.

Objetivo: Crear y modificar un esquema de base de datos para un sistema de ventas. El sistema debe gestionar la información de productos, clientes, ventas y los detalles de cada venta.

```
CREATE DATABASE SIS_VENTAS;

USE SIS_VENTAS;

-- Creo una nueva tabla de categorias ya que algunos productos
-- pertenecen a mas de una categoria
CREATE TABLE Categoria(
    ID_Categoria INT PRIMARY KEY,
    Nombre_categoria VARCHAR(100)
);

CREATE TABLE Producto (
    ID_Producto INT PRIMARY KEY,
    Nombre VARCHAR(255),
    Precio DECIMAL(10, 2)
);

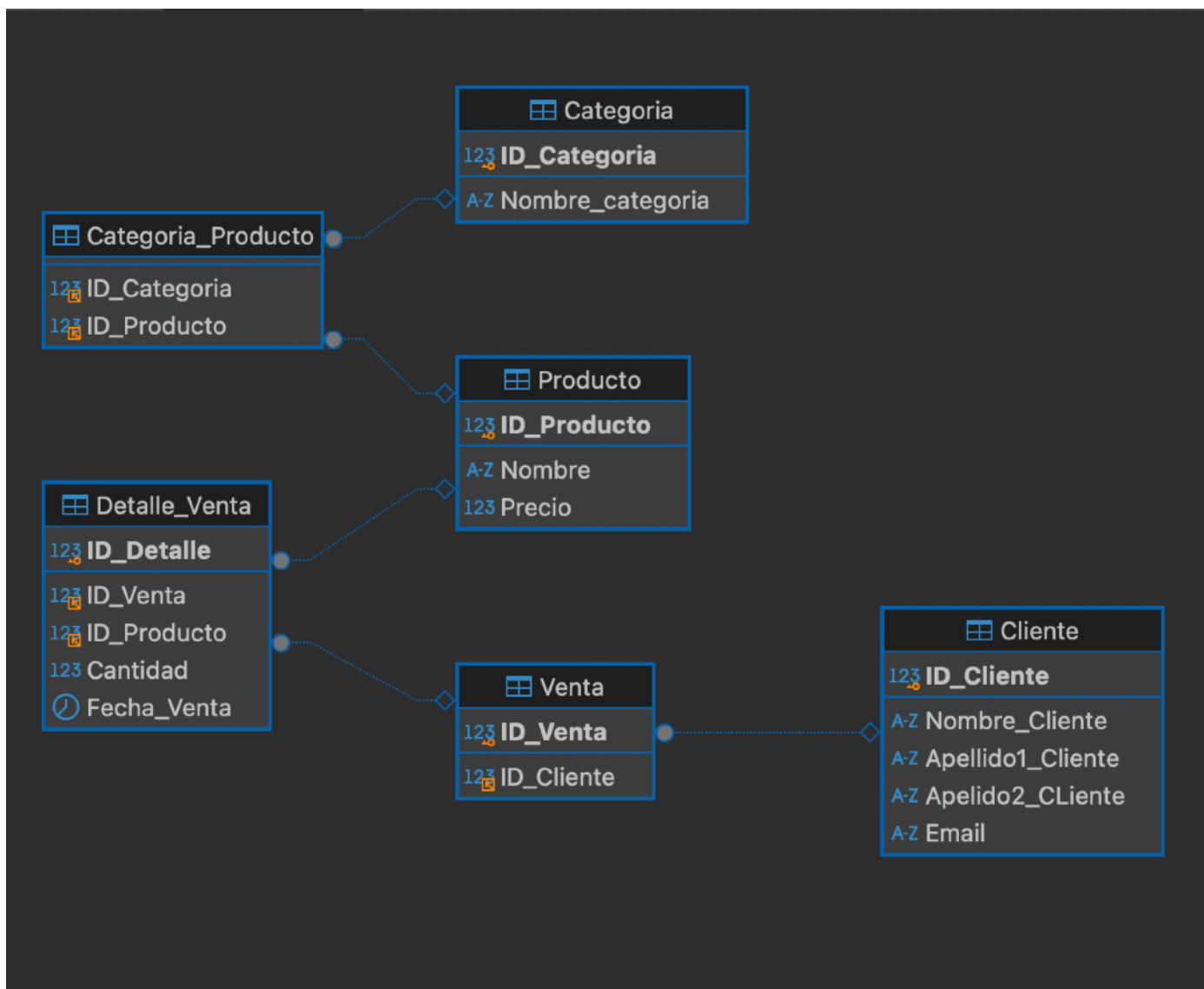
-- Cree una tabla de la relacion ya que la relacion de Categoria y Productos
-- Es de N:M
CREATE TABLE Categoria_Producto(
    ID_Categoria INT,
    ID_Producto INT,
    FOREIGN KEY (ID_Categoria) REFERENCES Categoria(ID_Categoria),
    FOREIGN KEY (ID_Producto) REFERENCES Producto(ID_Producto)
);

CREATE TABLE Cliente (
    ID_Cliente INT PRIMARY KEY,
    Nombre_Cliente VARCHAR(100),
    Apellido1_Cliente VARCHAR(100),
    Apellido2_Cliente VARCHAR(100),
    Email VARCHAR(255)
);

-- Aqui se modifiko ya que la fecha de venta va en detalle de venta
CREATE TABLE Venta (
    ID_Venta INT PRIMARY KEY,
    ID_Cliente INT,
    FOREIGN KEY (ID_Cliente) REFERENCES Cliente(ID_Cliente)
);

-- Se añadio el atributo fecha de venta
CREATE TABLE Detalle_Venta (
    ID_Detalle INT PRIMARY KEY,
    ID_Venta INT,
    ID_Producto INT,
    Cantidad INT,
    Fecha_Venta DATE,
    FOREIGN KEY (ID_Venta) REFERENCES Venta(ID_Venta),
    FOREIGN KEY (ID_Producto) REFERENCES Producto(ID_Producto)
);
```

Así queda el diagrama:





Problema 3: Diseñar y modificar un esquema para un sistema hospitalario.

Objetivo: Crear y modificar un esquema de base de datos para un sistema de hospital. El sistema debe gestionar la información de pacientes, médicos y citas.

```
CREATE DATABASE SIS_HOSPITAL;
USE SIS_HOSPITAL;

-- Aqui se modifiko el campo del nombre para que
-- no este en un solo registro y asi separado no hay redundancia
CREATE TABLE Paciente (
    ID_Paciente INT PRIMARY KEY,
    Nombre_paciente VARCHAR(100),
    Apellido1_paciente VARCHAR(100),
    Apellido2_paciente VARCHAR(100),
    Fecha_Nacimiento DATE,
    Sexo CHAR(1)
);

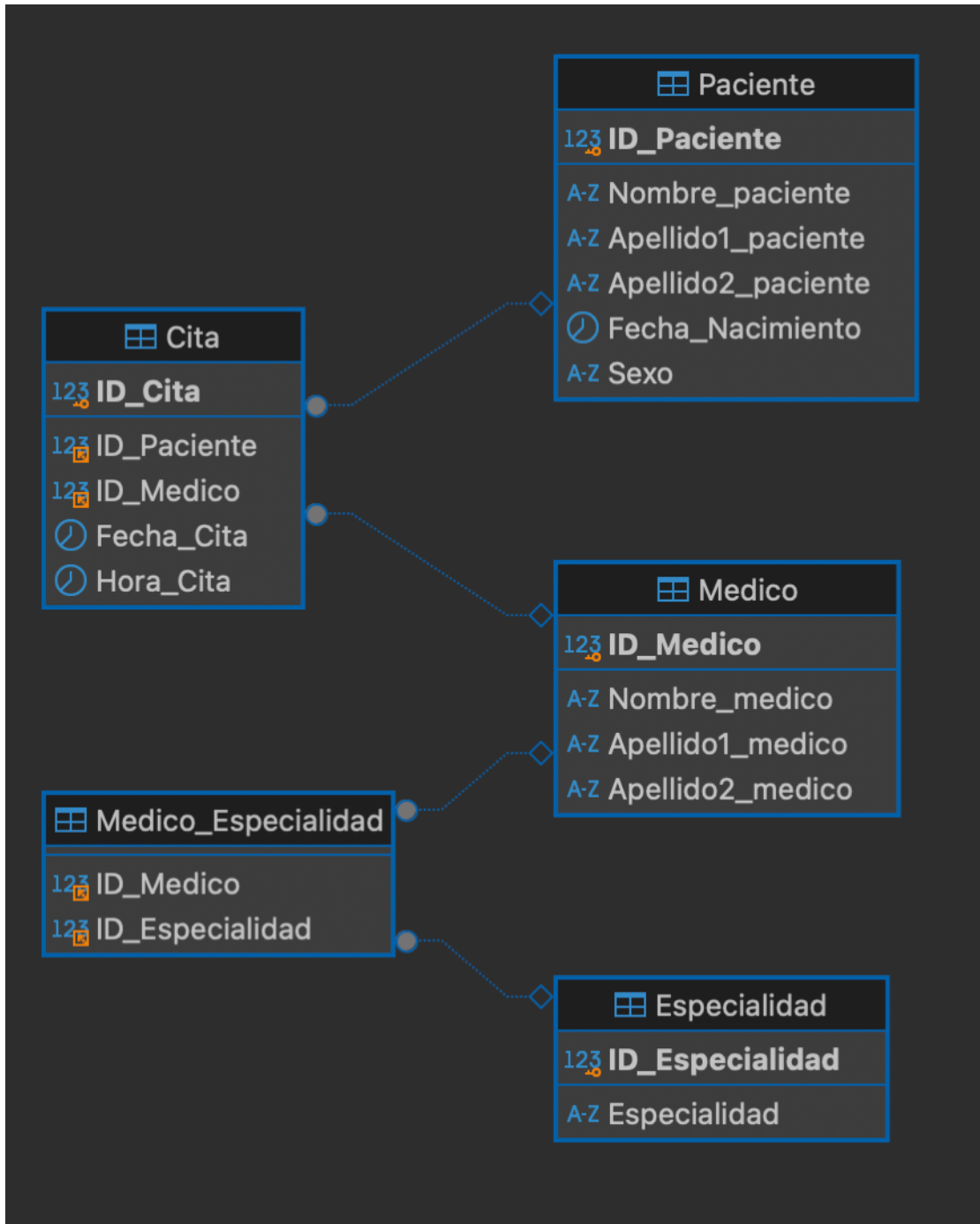
-- Aqui se modifiko el campo del nombre para que
-- no este en un solo registro y asi separado no hay redundancia
CREATE TABLE Medico (
    ID_Medico INT PRIMARY KEY,
    Nombre_medico VARCHAR(100),
    Apellido1_medico VARCHAR(100),
    Apellido2_medico VARCHAR(100)
);

-- Cree una tabla de Especialidad ya que algunas especialidades
-- existen para distintos medico y se puede repetir
CREATE TABLE Especialidad(
    ID_Especialidad INT PRIMARY KEY,
    Especialidad VARCHAR(100)
);

-- Como la relacion que se crea es de N:M se crea otra tabla con
-- los datos de las tablas
CREATE TABLE Medico_Especialidad(
    ID_Medico INT,
    ID_Especialidad INT,
    FOREIGN KEY (ID_Medico) REFERENCES Medico(ID_Medico),
    FOREIGN KEY (ID_Especialidad) REFERENCES Especialidad(ID_Especialidad)
);

CREATE TABLE Cita (
    ID_Cita INT PRIMARY KEY,
    ID_Paciente INT,
    ID_Medico INT,
    Fecha_Cita DATE,
    Hora_Cita TIME,
    FOREIGN KEY (ID_Paciente) REFERENCES Paciente(ID_Paciente),
    FOREIGN KEY (ID_Medico) REFERENCES Medico(ID_Medico)
);
```

Así queda el diagrama:



V. Conclusiones:

En esta practica aprendí a identificar los datos que suelen ser redundantes en las tablas y pues hay que corregirlos para evitar que estos de pasen al momento de la creación de la base de datos, también a tener mas practica a escribir en DDL ya que así es como funciona mysql y pues tener practica hace que cada vez uno mejore y creo yo que voy mejorando