**CORRECCIÓN FILTRO LIBRERÍA EL MUNDO DE SOFIA**

**ZULLY FERNANDA ORTIZ AVENDAÑO**

**DOCENTE**

**PEDRO FELIPE GÓMEZ BONILLA**

**CAMPUSLANDS**

**SANDBOX  
RUTA BASE DE DATOS**

**TIBU**

**2024**

[Ejecución 5](#_d5brfsc6k9x0)

[**Construcción del Modelo Conceptual 5**](#_cacjzqddwwvk)

[**Descripción 6**](#_gofm1qxci5rl)

[**Las entidades y atributos 6**](#_bb3qrnu96omb)

[**Relaciones y Cardinalidades 7**](#_zgn1tkor45d0)

[**Descripción 11**](#_m8vzqdhmspp)

[**Descripción técnica 11**](#_knailrb2yve0)

[**Descripción 12**](#_6py2m1mxdct7)

[**Descripción Técnica 12**](#_xafy1nwtcn1e)

[**Descripción Técnica 14**](#_88m1wvcw5wg)

INTRODUCCIÓN

En este proyecto se presentará una guía detallada de la estructura e implementación de una base de datos para la librería del mundo de Sofía, donde el objetivo principal es gestionar un inventario de ventas y clientes.

Primero se realizará un caso de estudio con sus requerimientos específicos, donde a partir de estos análisis se procederá a desarrollar un modelo conceptual detallado donde se identificará las entidades principales, sus atributos y relaciones de todas ellas, donde este paso es realizado para comprender la estructura que se llevará a cabo para la librería.

Prosiguiendo se realizará la conversión del modelo conceptual al lógico, donde este tiene como función representar de manera más concisa de la organización de toda la base de datos, facilitando si compression, también se aplicará el proceso de normalización hasta la tercera forma normal (3FN) donde se reducira los datos redundantes.

Finalmente se llevará a cabo la conversión del modelo lógico al modelo físico, el cual definirá las entidades, atributos y relaciones.

Con estos pasos se garantizara un proyecto completo y efectivo para el diseño de los datos de datos para la librería.

CASO DE ESTUDIO

La librería del mundo de sofía nos permitirá crear una base de datos donde deberá gestionar todo el inventario, las ventas y los clientes, donde se debe sacar cada registro para poder tener la información de la gestión de los libros, autores, clientes, pedidos y transacciones de compra, para lograr eso creare un diagrama UML E-R donde se quedará evidenciado toda la base de datos de la tienda, por lo que tendrá los siguientes requerimientos para la elaboración:

* Libros:

Un libro tiene un título, uno o más autores, editorial, categoría, fecha de publicación, ISBN único, precio y cantidad en stock.

Permitir buscar libros por título, autor, categoría o ISBN.

* Autores:

Un autor tiene un nombre, fecha de nacimiento y nacionalidad.

Un libro puede tener varios autores y un autor puede haber escrito varios libros.

* Clientes:

Los clientes se registran con nombre, correo electrónico, teléfono y dirección.

Un cliente puede realizar varios pedidos.

* Pedidos:

Un pedido puede incluir múltiples libros, con cantidades específicas.

Asociar cada pedido a un cliente y una fecha de compra.

Gestionar el estado del pedido (pendiente, procesado, completado).

* Transacciones:

Cada pedido genera una transacción con el método de pago (tarjeta de crédito, PayPal, etc.), el monto total y la fecha de la transacción.

* Relaciones y Restricciones:

Implementar relaciones entre libros y autores, entre clientes y pedidos, y entre pedidos y transacciones.

Evitar registros duplicados de libros.

La cantidad en stock de un libro debe actualizarse al realizar una compra.

**PLANIFICACIÓN**

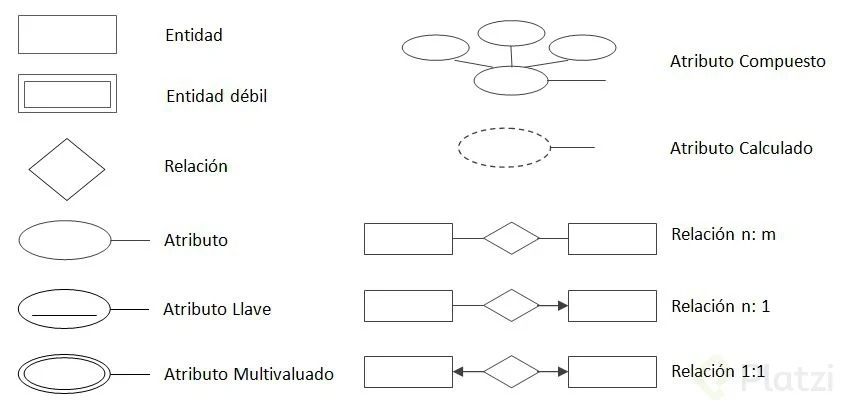
### 

### Ejecución

Una vez haya sido analizado toda la información requerida para la elaboración de base de datos de la librería el mundo de Sofía, se inició la elaboración del modelo conceptual, la cual este proceso proporciona un alta descripción donde representará los conceptos principales de la base de datos y las relaciones que hay entre ellos.

## Construcción del Modelo Conceptual

Se realizó el Modelo conceptual para identificar cada una de las entidades, sus atributos y las relaciones entre ellas, donde se proporcionó una visión más clara y estructurada de cómo se organizó y conecto los diferentes modelos de la base de dato teniendo en cuenta lo siguiente:



## Descripción

### Las entidades y atributos

1. Libros:

* Id\_libros.
* Titulo.
* Autores.
* Editorial.
* Categoría.
* Fecha\_publicacion.
* ISB.
* precio.
* stock.

1. Autores:

* Id\_autores.
* nombre.
* Fecha\_nacimiento.
* nacionalidad.

1. Clientes:

* Id\_clientes.
* Nombre.
* Correo\_electronico.
* Telefono.
* Direccion.

1. Pedidos:

* Id\_pedido.
* Id\_libro.
* Cantidad\_stock.
* Fecha\_compra.
* estado\_pedido.

1. Transacciones:

* Id\_transacciones.
* Metodo\_pago.
* Monto\_total.
* Fecha\_transaccion.

### Relaciones y Cardinalidades

1. Autores - Libros:

* Relación: “Tiene”, Un autor puede tener varios libros y varios libros pueden tener un autor.
* Cardinalidad: 1-N (uno a muchos).

1. Clientes - Libros:

* Relación: “Alquilar”, Un cliente puede alquilar varios libros y varios libros pueden ser alquilados por un cliente.
* Cardinalidad: 1:N (uno a muchos).

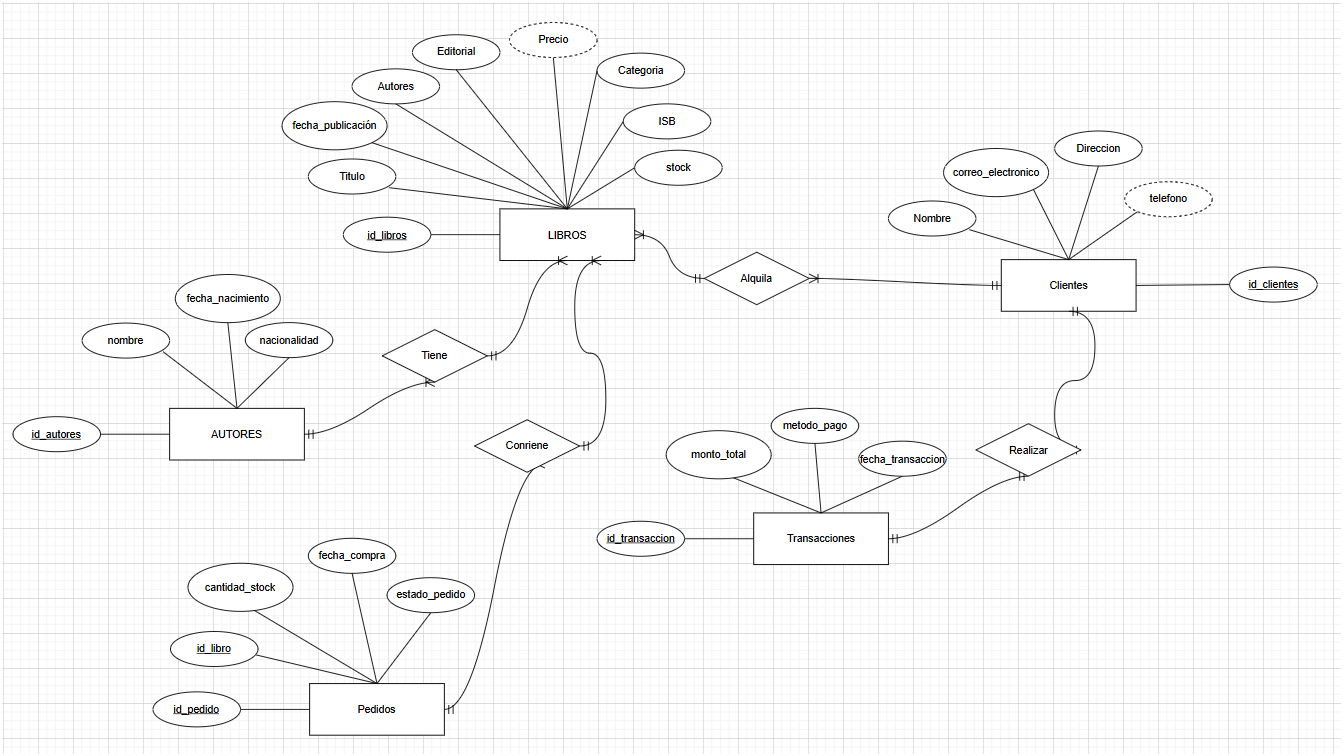
1. Clientes - Transacciones:

* Relación: “Realizar”, Un cliente puede realizar una transacción y una transacción podría ser realizada por un cliente.
* Cardinalidad: 1:1 (uno a uno).

1. Pedido - libro:

* Relación: ‘Contener’, Un pedido contiene varios libros y varios libros pueden estar contenido en un pedido.
* Cardinalidad: 1:N (uno a muchos).

**Gráfica**

****

Diseño completo del modelo conceptual en draw.io

**Construcción del Modelo Lógico**

Se realizó el modelo lógico teniendo en cuenta el modelo conceptual, donde se incorporaron detalles más específicos como las características de cada atributo, incluidas las claves primarias, foráneas y las relaciones de cardinalidad.

Descripción

1. Libros:

* Id\_libros: int(20) unique not null
* Titulo: varchar(50) not null
* Autores: varchar(100)not null
* Editorial: varchar(100)not null
* categoría: varchar(50)not null
* Fecha\_publicacion: DATE
* ISB: int(20)not null
* Precio. int(50)not null
* Stock. int(50)not null

1. Autores:

* id\_autores: int(20)unique not null primary key
* nombre: varchar(30)not null
* Fecha\_nacimiento:date
* nacionalidad:varchar(30)not null

1. Clientes:

* Id\_cliente:int(30)unique not null primary key
* Nombre: varchar(30)not null
* Correo\_electronico:varchar(30) not null
* Teléfono: int (30) not null
* Dirección : varchar(30)not null

1. Transacciones:

* Id\_transacciones:int(30)unique not null primary key
* Método\_pago:varchar(30)not null
* Monto\_total:varchar (30)not null
* Fecha\_transacion:date

1. Pedidos

* Id\_pedidos:int(30)not null primary key
* Id\_libros:int(30)not null foreign key
* Cantidad\_stock:int(30)not null
* Fecha\_compra:date
* Estado\_pedido: varchar(50)

**Relaciones y Cardinalidades**

Se realizó las relaciones y cardinalidades respectivas del modelo lógico con sus entidades para tener mejor visualización de la base de datos:

1. Autores - Libros:

* Un autor puede tener varios libros y varios libros pueden tener un autor. 1-N (uno a muchos).



1. Clientes - Libros:

* Un cliente puede alquilar varios libros y varios libros pueden ser alquilados por un cliente.1:N (uno a muchos).



1. Clientes - Transacciones:

* Un cliente puede realizar una transacción y una transacción podría ser realizada por un cliente. 1:1 (uno a uno).

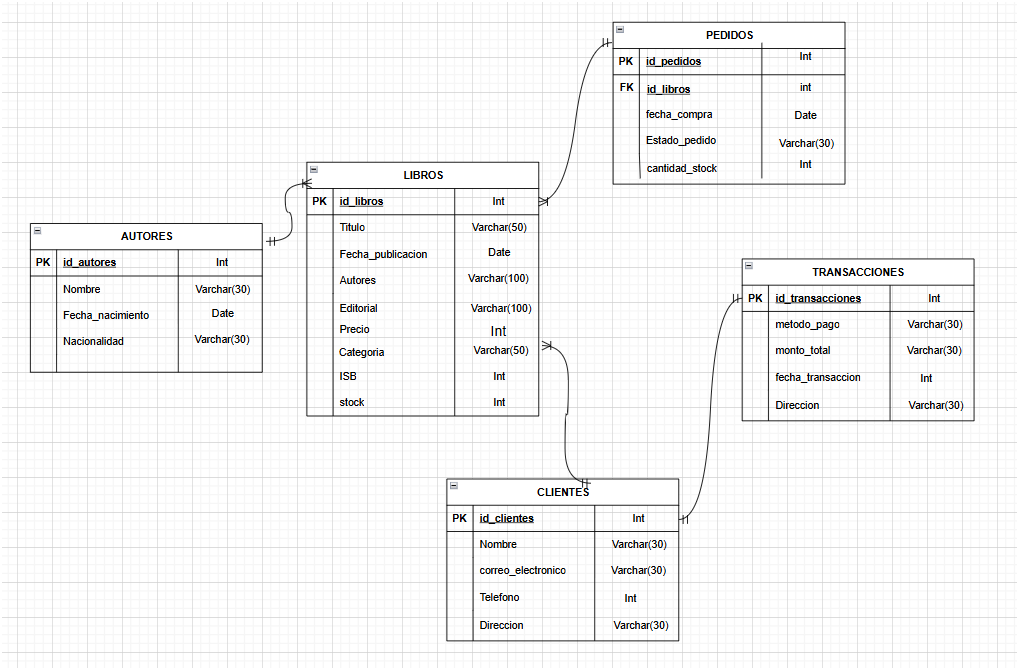


1. Pedido - libro:

* Un pedido contiene varios libros y varios libros pueden estar contenido en un pedido.1:N (uno a muchos).



**Gráfica**

****

Diseño completo del modelo logico en draw.io

Normalización del Modelo Lógico

Se realizó el proceso de la normalización de las tablas anteriormente visualizadas para

organizar los datos de manera más eficiente,minimizando redundancias y dependencias

transitivas en la base de datos en desarrollo.

Primera Forma Normal (1FN)

Una tabla está en 1FN si cumple con los siguientes criterios:

* Cada atributo debe tener valores atómicos.
* Cada fila la misma tabla debe ser única
* Debe prevalecer un crecimiento vertical de los datos y no horizontal
* No deben existir grupos repetidos de datos.

### Descripción

La primera forma normal es el primer nivel de normalización en base de datos donde se le aplicará a las tablas de las bases de datos para garantizar una mejor organización donde se evita redundancias y asegurará la consistencia de la información.

### Descripción técnica

1. Libros :

❖ Se encuentra en 1FN, ya que cuenta con una clave primaria única y cada

La columna tiene valores únicos y no son repetitivos.

1. Autores :

❖ Se encuentra en 1FN, ya que cuenta con una clave primaria única y cada

La columna tiene valores únicos y no son repetitivos.

3. Clientes :

❖ Se encuentra en 1FN, ya que cuenta con una clave primaria única y cada

La columna tiene valores únicos y no son repetitivos.

4. Transacciones :

❖ Se encuentra en 1FN, ya que cuenta con las claves primarias compuestas y

Cada columna tiene valores únicos y no son repetitivos.

5. Pedidos:

❖ Se encuentra en 1FN, ya que cuenta con las claves primarias compuestas y

Cada columna tiene valores únicos y no son repetitivos.

Gráfica

| Libros | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_libros | titulo | autores | editorial | categoria | fecha\_publicacion | ISB | precio | stock |
| PK |  |  |  |  |  |  |  |  |

|  |  | Autores |  |  |
| --- | --- | --- | --- | --- |
| id\_autores | nombre | correo\_electronico | telefono | direccion |
| PK |  |  |  |  |

| Clientes | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_clientes |  | nombre | correo\_electronico |  | telefono | direccion |
| PK | |  |  | |  |  |

| Transacciones | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_transacciones |  | metodo\_pago |  | monto\_total |  | fecha\_transacion |  |
| PK | |  | |  | |  | |

| Pedidos | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_pedidos |  | id\_libros | cantidad\_stock |  | fecha\_compra |  | estado\_pedido |
| PK | | FK |  | |  | |  |

Segunda Forma Normal (2FN)

Una tabla está en 2FN si cumple con los siguientes criterios:

* Estar en 1FN.
* La relación debe tener una clave principal, de preferencia simple.
* Cada atributo de la tabla debe depender totalmente del atributo clave.

### Descripción

La segunda forma normal, es el segundo nivel de normalización en el diseño de la base de

datos que se aplicará a las tablas de una base de datos que ya cumplen con la primera forma

normal y lleva a cabo la eliminación de dependencias parciales dentro de una tabla.

### Descripción Técnica

1. Libros :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Autores :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Clientes :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Transacciones :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Pedidos:

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

| Libros | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_libros | titulo | autores | editorial | categoria | fecha\_publicacion | ISB | precio | stock |
| PK |  |  |  |  |  |  |  |  |

|  |  | Autores |  |  |
| --- | --- | --- | --- | --- |
| id\_autores | nombre | correo\_electronico | telefono | direccion |
| PK |  |  |  |  |

| Clientes | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_clientes |  | nombre | correo\_electronico |  | telefono | direccion |
| PK | |  |  | |  |  |

| Transacciones | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_transacciones |  | metodo\_pago |  | monto\_total |  | fecha\_transacion |  |
| PK | |  | |  | |  | |

| Pedidos | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_pedidos |  | id\_libros | cantidad\_stock |  | fecha\_compra |  | estado\_pedido |
| PK | | FK |  | |  | |  |

Tercera Forma Normal (3FN)

Una tabla está en 3NF si cumple con los siguientes criterios:

* Debe estar en 2FN.
* No deben existir atributos no principales que dependen transitivamente del atributo clave.

Descripción

La tercera forma normal, es el tercer nivel de normalización en el diseño de la base de datos

que se aplicará a las tablas de una base de datos que ya cumplen con la segunda forma

normal y se enfoca en la eliminación de dependencias transitivas, evitando que un atributo no

clave depende de otro no clave.

### Descripción Técnica

1. Libros :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Autores :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Clientes :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Transacciones :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Pedidos:

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

| Libros | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_libros | titulo | autores | editorial | categoria | fecha\_publicacion | ISB | precio | stock |
| PK |  |  |  |  |  |  |  |  |

|  |  | Autores |  |  |
| --- | --- | --- | --- | --- |
| id\_autores | nombre | correo\_electronico | telefono | direccion |
| PK |  |  |  |  |

| Clientes | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_clientes |  | nombre | correo\_electronico |  | telefono | direccion |
| PK | |  |  | |  |  |

| Transacciones | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_transacciones |  | metodo\_pago |  | monto\_total |  | fecha\_transacion |  |
| PK | |  | |  | |  | |

| Pedidos | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id\_pedidos |  | id\_libros | cantidad\_stock |  | fecha\_compra |  | estado\_pedido |
| PK | | FK |  | |  | |  |

Construccion del Modelo Fisico

Se realizó el modelo físico acorde con la elaboración del modelo lógico donde incluye las entidades, sus atributos y las relaciones entre ellas. El objetivo de este modelo es incorporar los tipos de datos de los atributos previamente definidos.

Descripción

El modelo físico se trabaja en MySQL, en el cual cada entidad se representa como una tabla compuesta por sus atributos correspondientes, organizados en columnas con tipos de datos específicos según sea necesario.

Tablas

Para crear la base de datos utilice el siguiente comando:



Para utilizar la base de datos ocupe el siguiente comando:



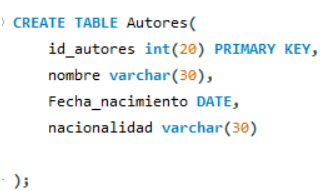
Comenzaremos creando las tablas junto con sus tipos de datos correspondientes. Para esto,

utiliza los siguientes comandos:

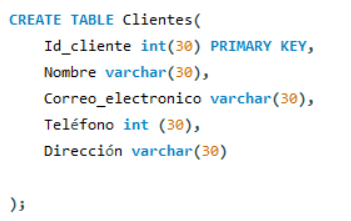
1. Creación de la tabla Libros:



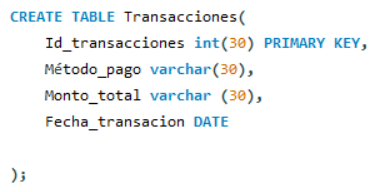
2. Creación de la tabla Autores:



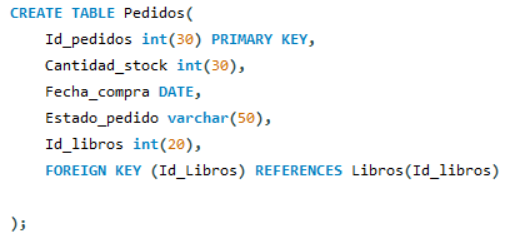
3. Creación de la tabla Clientes:



4. Creación de la tabla Transacciones:



5. Creación de la tabla Pedidos:



Construcción del Diagrama UML

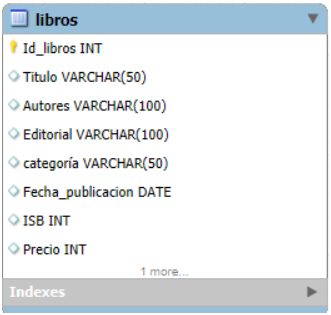
Se realizó un diagrama UML tomando como referencia la normalización para poder entender mejor cada diseño, la arquitectura del código y la implementación. Dónde nos permitirá tener una visión clara y detallada de cómo se maneja cada consulta de la librería.

Descripción

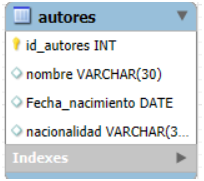
El diagrama UML se diseñó con el objetivo de representar detalladamente la estructura de cada tabla y sus relaciones, donde tendrá así más ganadas unas llaves primarias y foráneas, además especifica obligatoriamente los atributos, dando un nivel de detalle facilitada para poder entenderse con más facilidad.

Comenzaremos creando las tablas junto con sus tipos de datos correspondientes:

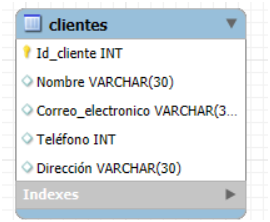
1.Tabla Libros:



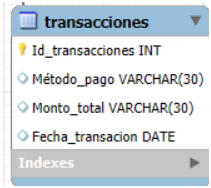
2. Tabla Autores:



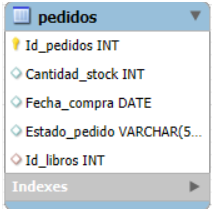
3. Tabla Clientes:



4. Tabla Transacciones:



5. Tabla Pedidos:



Grafica

