



Introducción a las bases de datos y arquitectura de un DBMS.

2024

CONTENIDO

1. Programa y objetivos
2. Introducción a las Bases de Datos y DBMS
3. ¿Qué es una base de datos?
4. Evolución de las bases de datos
5. Rol de los DBMS
6. ¿Qué es SQL?
7. Modelado de Base de datos
8. Ejercicios
9. Cierre y conclusión

Programas y Objetivos

1 .Introducción a las bases de datos y arquitectura de un DBMS.	0	12	N/A	0	Otros	0	0	0	4	16
2 .Modelamiento de bases de datos.	0	18	N/A	0	Otros	0	0	0	6	24
3 .Definición, manipulación y consulta de datos usando SQL.	0	24	N/A	0	Otros	0	0	0	8	32



Introducción a las Bases de Datos y DBMS

Comprender el papel fundamental de las bases de datos en las organizaciones modernas.

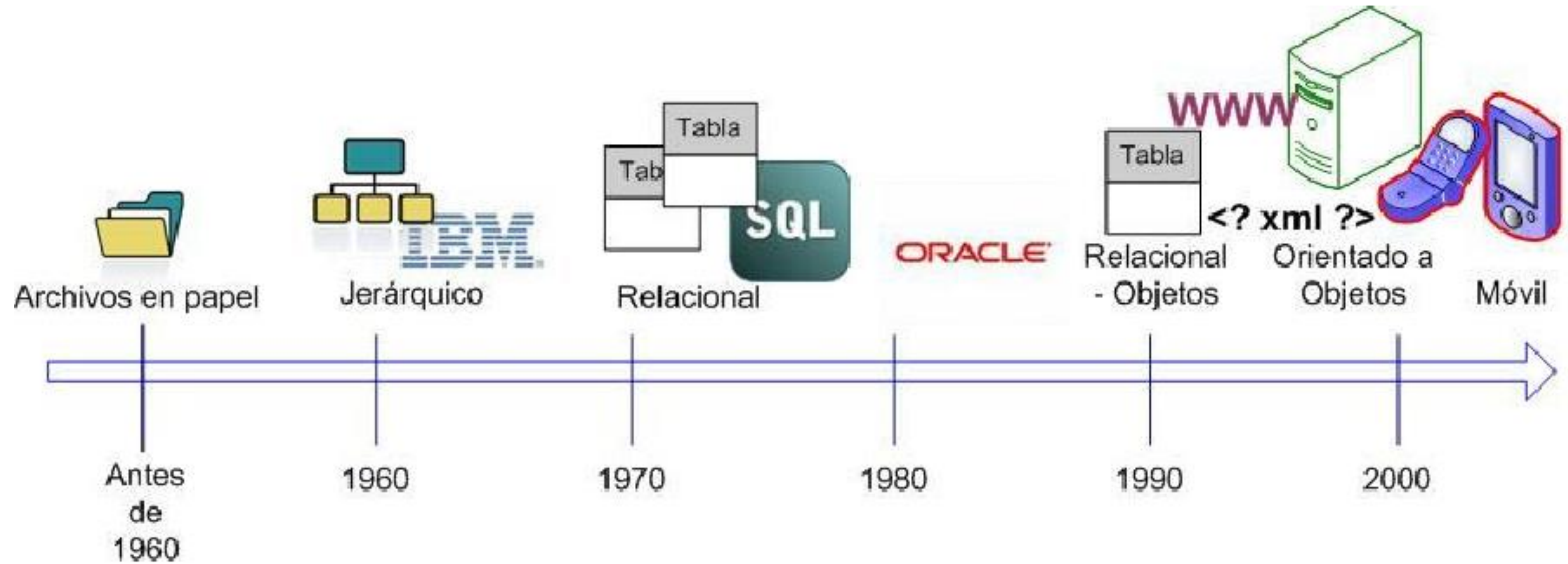


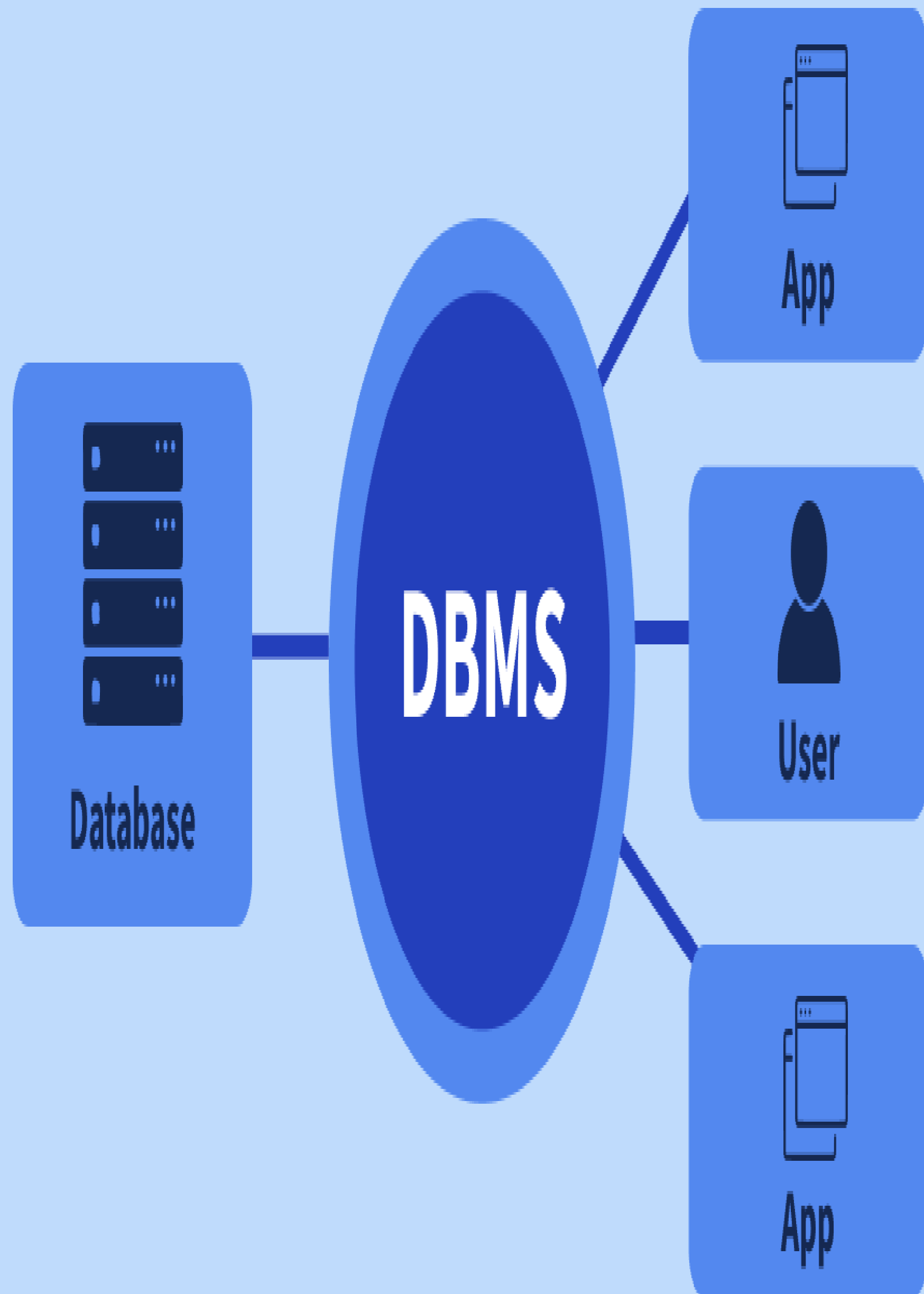
¿Qué es una base de datos?

Una base de datos es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un sistema de gestión de bases de datos (DBMS).



Evolución de las Bases de Datos





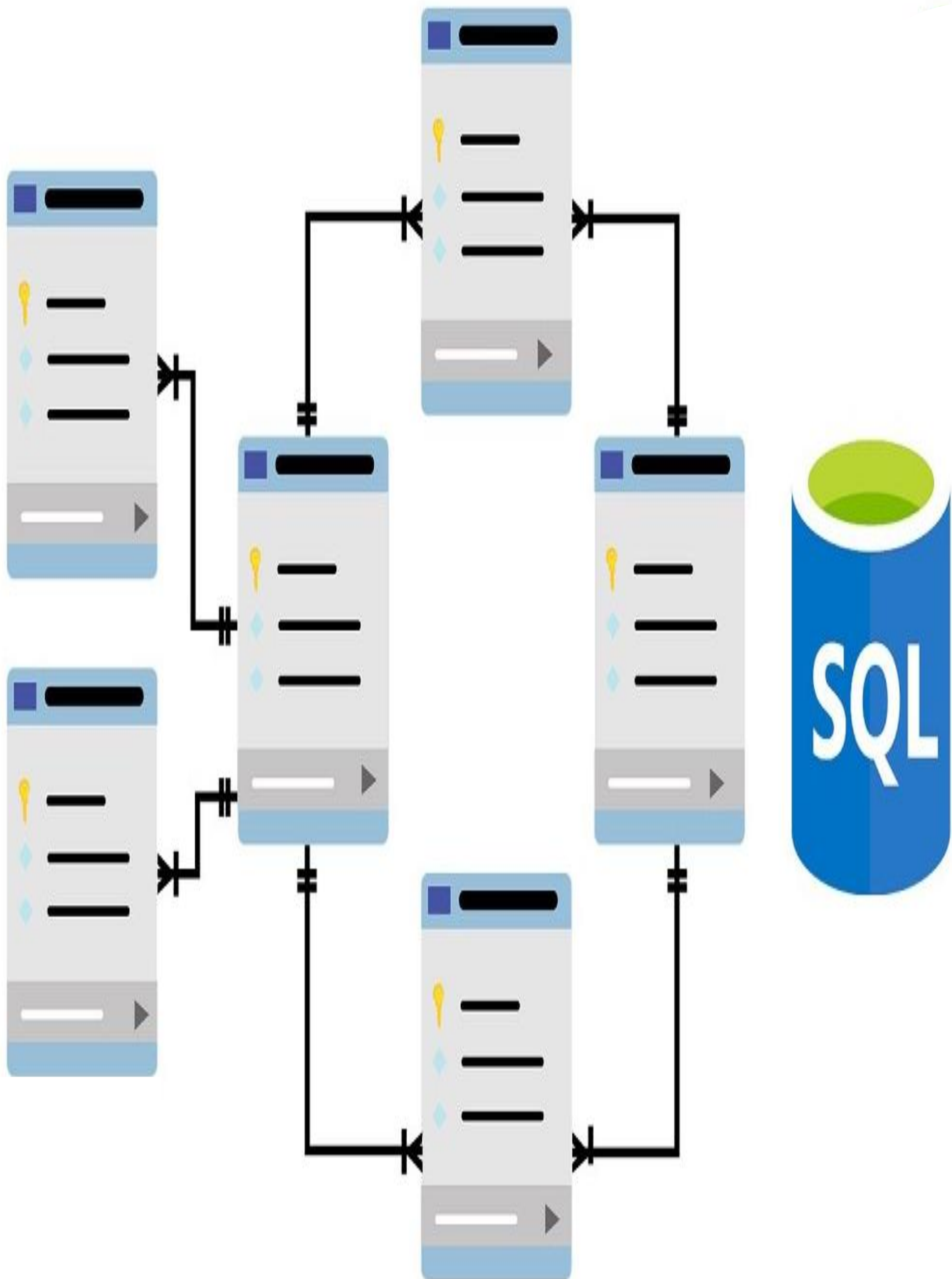
Rol de los DBMS

Los Sistemas de Gestión de Bases de Datos (DBMS) son software diseñado para permitir la creación, manipulación y administración eficiente de bases de datos.

Concepto de Dato e Información

- Los datos son hechos básicos o detalles sobre algo, mientras que la información es el resultado del procesamiento, manipulación y organización de esos datos para revelar significado.





¿Qué es SQL?

El lenguaje de consulta estructurada (SQL) es un lenguaje de programación para almacenar y procesar información en una base de datos relacional.

8:15 PM



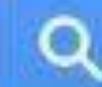
como borrar registros de una tabla



8:20 PM



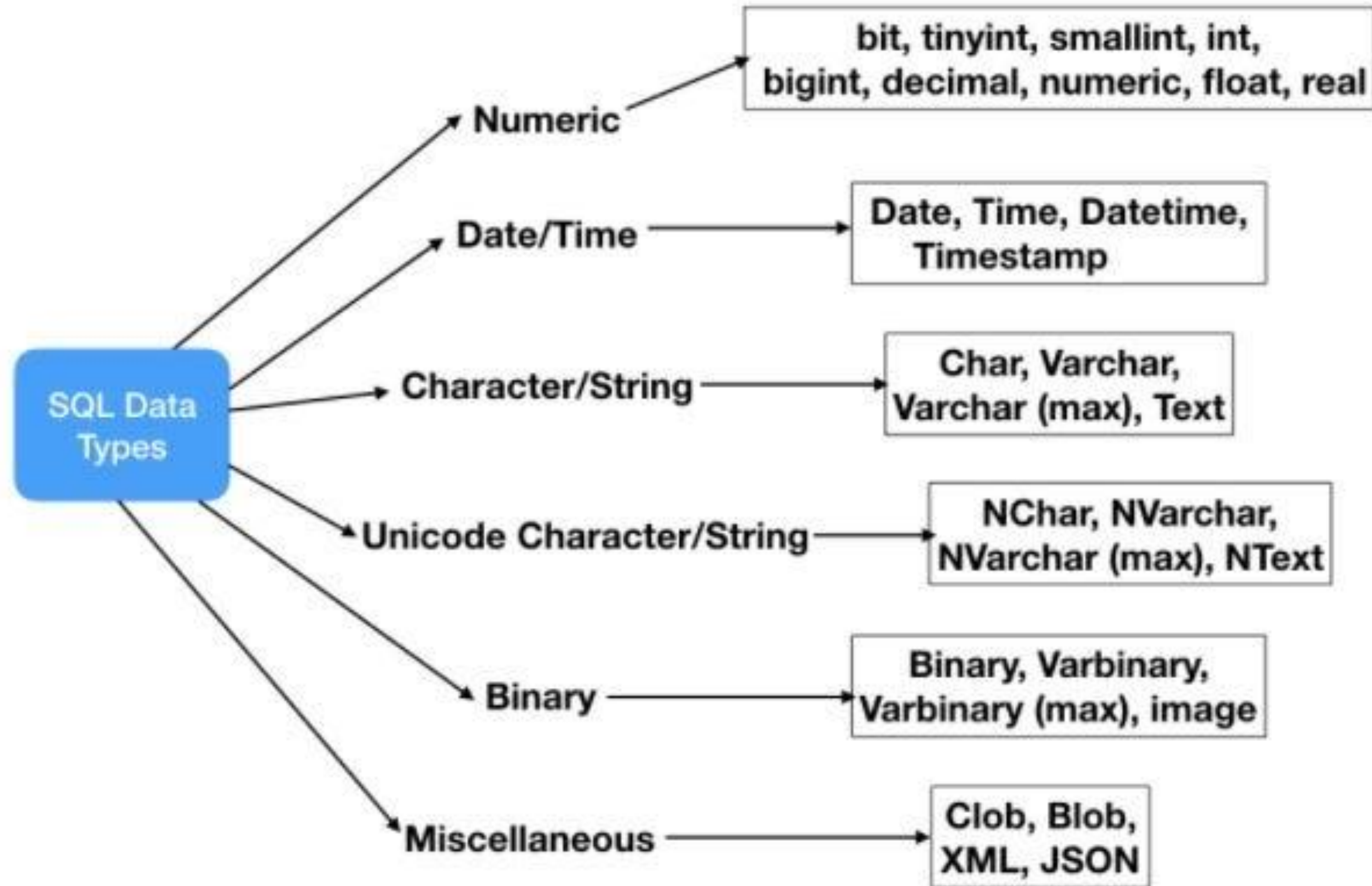
como recuperar base de datos borrada



Comandos básicos para SQL



Tipos de datos en SQL



Sintaxis SQL

Indica las columnas
que se quieren consultar.

Selecciona las tablas.

Define las condiciones para
que se cumpla la consulta.

Muestra cómo se agruparán
los registros. Se usa **Group
by** cuando existen funciones
como **COUNT()**.

Indica qué condición debe
cumplirse como resultado.
*COUNT() cuenta los libros y la regla
determina que sean mayores a 3.*

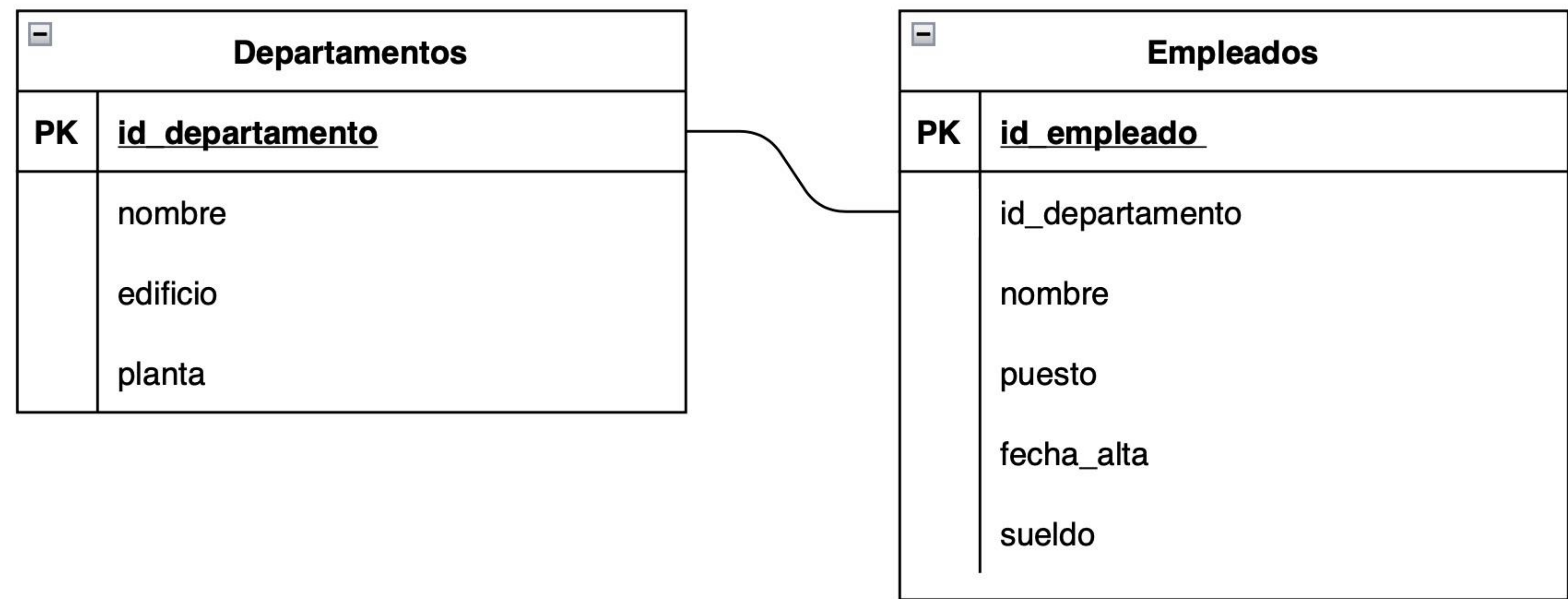
Muestra el orden de la consulta.

```
SELECT nombre, autor, genero
FROM libros
WHERE genero = 'Novela negra';

SELECT autor, COUNT(libroID)
FROM libros
WHERE genero = 'Novela negra'
GROUP BY autor
HAVING COUNT(libroID) > 3
ORDER BY nombre ASC;
```


Modelado de Base de datos

El modelado de datos es el proceso de diagramación de los flujos de datos. Al crear la estructura de una base de datos nueva o alternativa, el diseñador comienza con un diagrama del flujo de los datos por dentro y fuera de la base de datos.



Ejercicios de modelado 1

Descripción: Imaginemos que estamos diseñando la base de datos para una tienda de libros en línea. La tienda necesita realizar un seguimiento de los libros disponibles, los autores y los clientes que realizan compras.

Entidades:

- Libro
- Autor
- Cliente
- Compra

Atributos:

Libro:

- ID (Identificador único del libro)
- Título (Título del libro)
- Autor_ID (Identificador único del autor del libro)
- Precio (Precio del libro)

Autor:

- ID (Identificador único del autor)
- Nombre (Nombre del autor)
- Nacionalidad (Nacionalidad del autor)

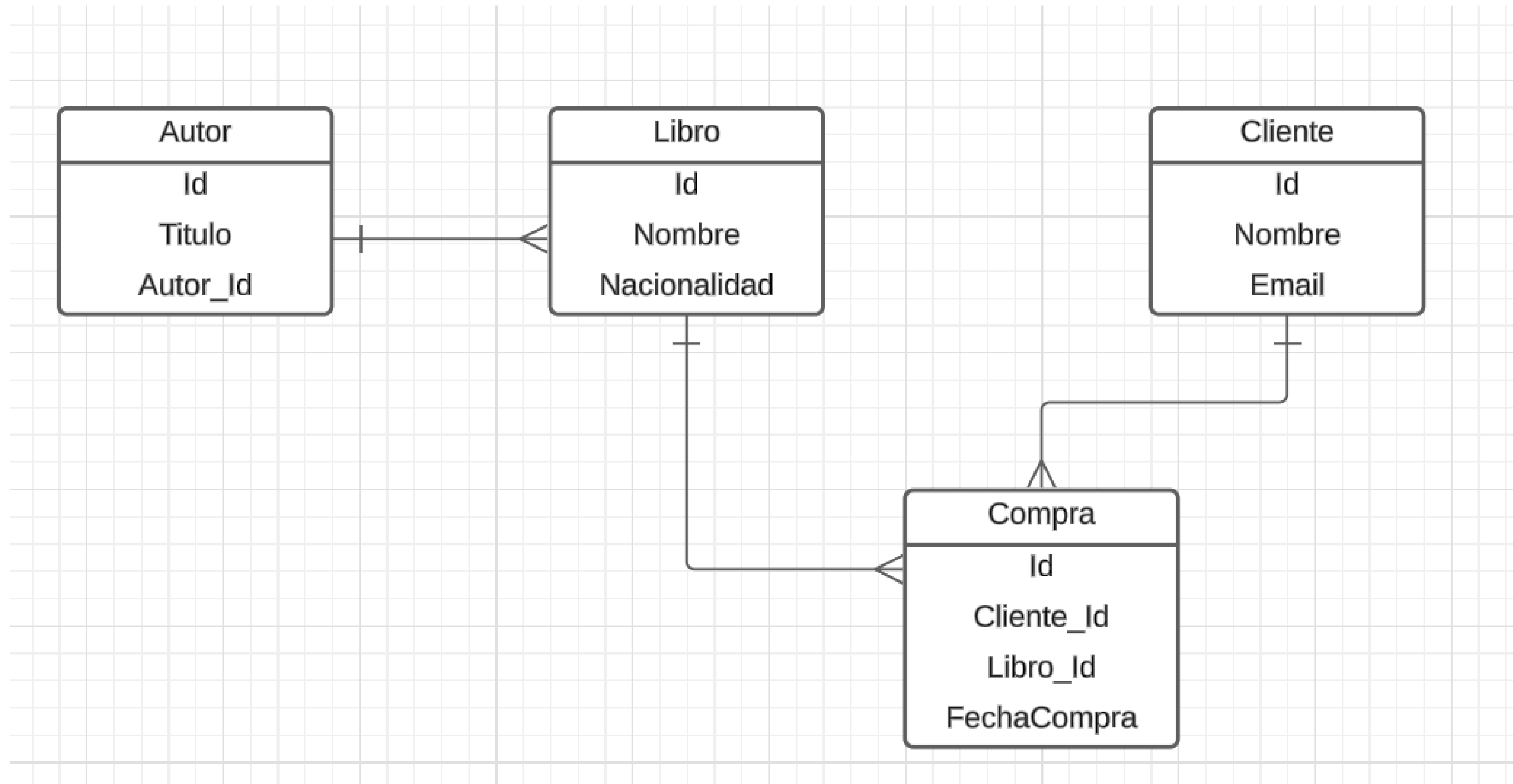
Cliente:

- ID (Identificador único del cliente)
- Nombre (Nombre del cliente)
- Email (Correo electrónico del cliente)
- Dirección (Dirección del cliente)

Compra:

- ID (Identificador único de la compra)
- Cliente_ID (Identificador único del cliente que realizó la compra)
- Libro_ID (Identificador único del libro comprado)
- Fecha_Compra (Fecha en que se realizó la compra)

Solución Modelado 1



Ejercicios de modelado 2

Descripción: Supongamos que estamos diseñando la base de datos para una tienda de productos electrónicos en línea. La tienda necesita realizar un seguimiento de los productos disponibles, las marcas, los clientes y las órdenes de compra.

Entidades:

- 1.Producto
- 2.Marca
- 3.Cliente
- 4.Orden

Atributos:

1.Producto:

- 1. ID (Identificador único del producto)
- 2. Nombre (Nombre del producto)
- 3. Descripción (Descripción del producto)
- 4. Precio (Precio del producto)
- 5. Marca_ID (Identificador único de la marca del producto)

2.Marca:

- 1. ID (Identificador único de la marca)
- 2. Nombre (Nombre de la marca)
- 3. País (País de origen de la marca)

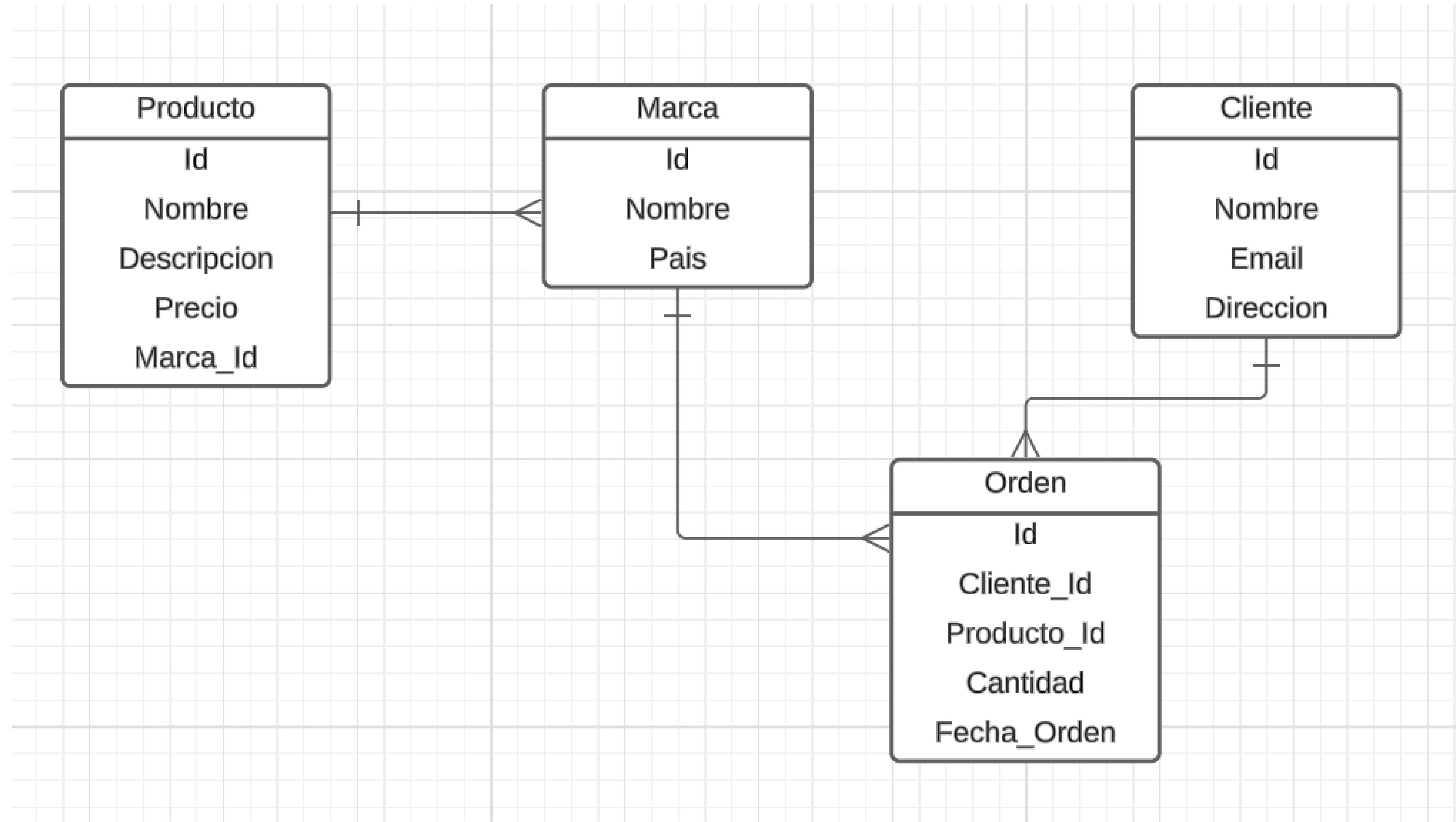
3.Cliente:

- 1. ID (Identificador único del cliente)
- 2. Nombre (Nombre del cliente)
- 3. Email (Correo electrónico del cliente)
- 4. Dirección (Dirección del cliente)

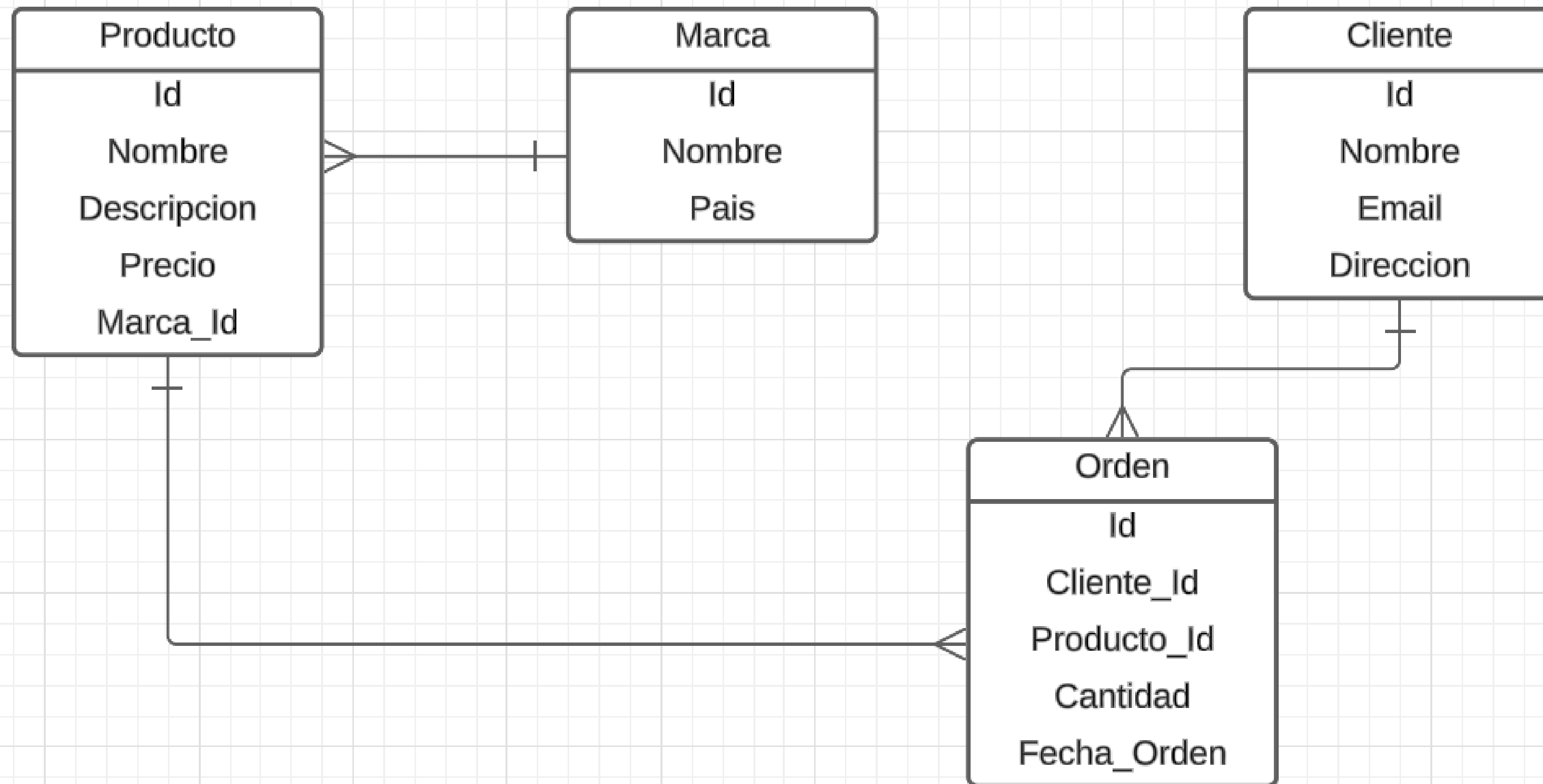
4.Orden:

- 1. ID (Identificador único de la orden)
- 2. Cliente_ID (Identificador único del cliente que realizó la orden)
- 3. Producto_ID (Identificador único del producto ordenado)
- 4. Cantidad (Cantidad de productos ordenados)
- 5. Fecha_Orden (Fecha en que se realizó la orden)

Solución Modelado 2



Solución Modelado 2



Ejercicios de modelado 3

Descripción: Supongamos que estamos diseñando la base de datos para un restaurante. El restaurante necesita realizar un seguimiento de los platos disponibles en el menú, los ingredientes utilizados en cada plato, los clientes que realizan pedidos y los pedidos realizados.

Entidades:

- Plato
- Ingrediente
- Cliente
- Pedido

Atributos:

Plato:

- ID (Identificador único del plato)
- Nombre (Nombre del plato)
- Descripción (Descripción del plato)
- Precio (Precio del plato)

Ingrediente:

- ID (Identificador único del ingrediente)
- Nombre (Nombre del ingrediente)
- Cantidad (Cantidad disponible del ingrediente)

Cliente:

- ID (Identificador único del cliente)
- Nombre (Nombre del cliente)
- Teléfono (Número de teléfono del cliente)
- Dirección (Dirección del cliente)

Pedido:

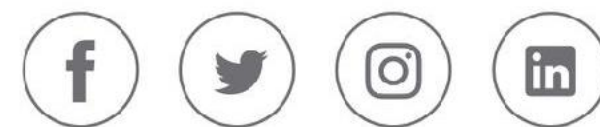
- ID (Identificador único del pedido)
- Cliente_ID (Identificador único del cliente que realizó el pedido)
- Plato_ID (Identificador único del plato pedido)
- Cantidad (Cantidad de platos pedidos)
- Fecha_Pedido (Fecha en que se realizó el pedido)

Ejercicios



<https://github.com/ebravo930/claseBaseDatos01>

MUCHAS GRACIAS!



inacap.cl