



Clase 6 : Bases de datos

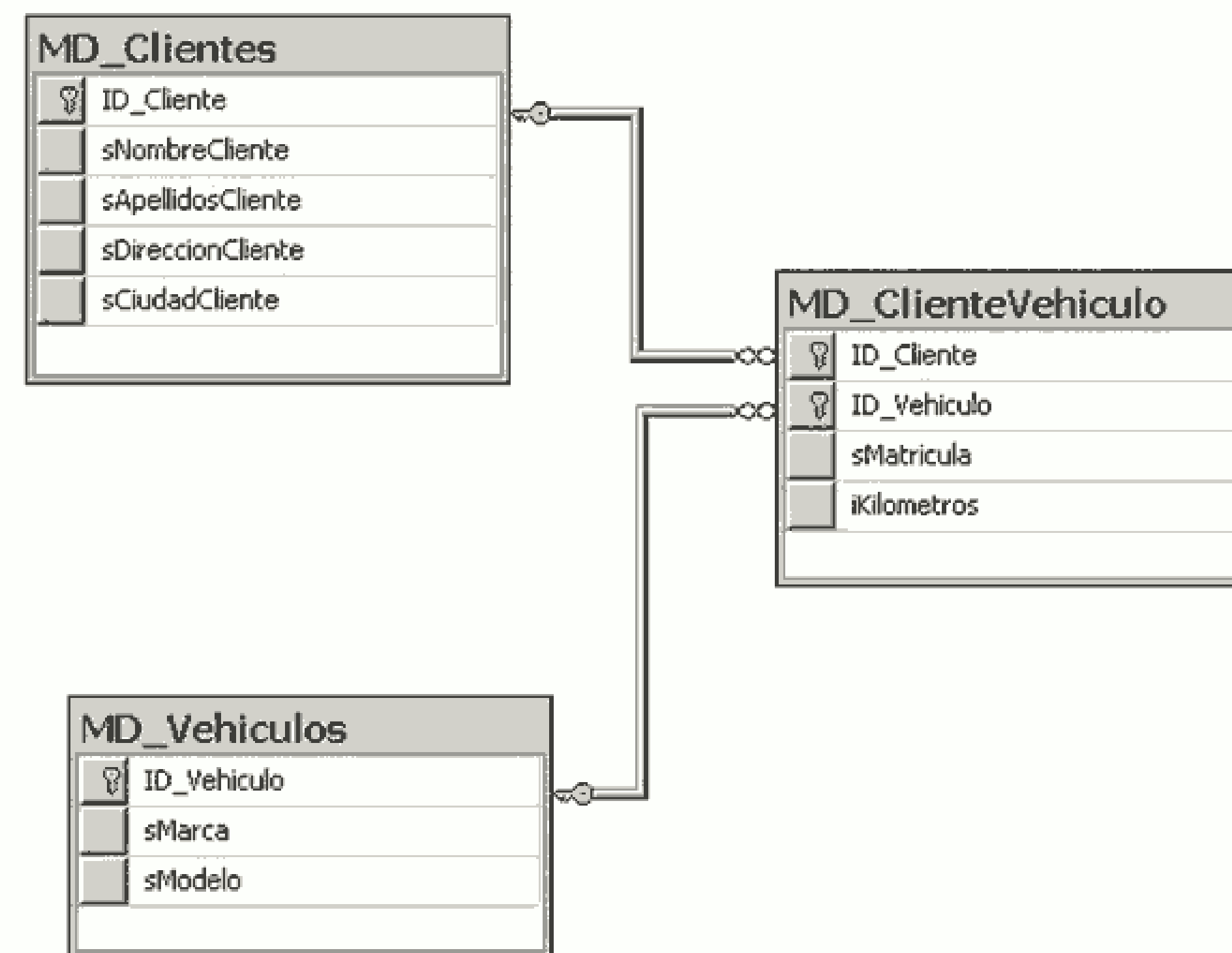
2024

CONTENIDO

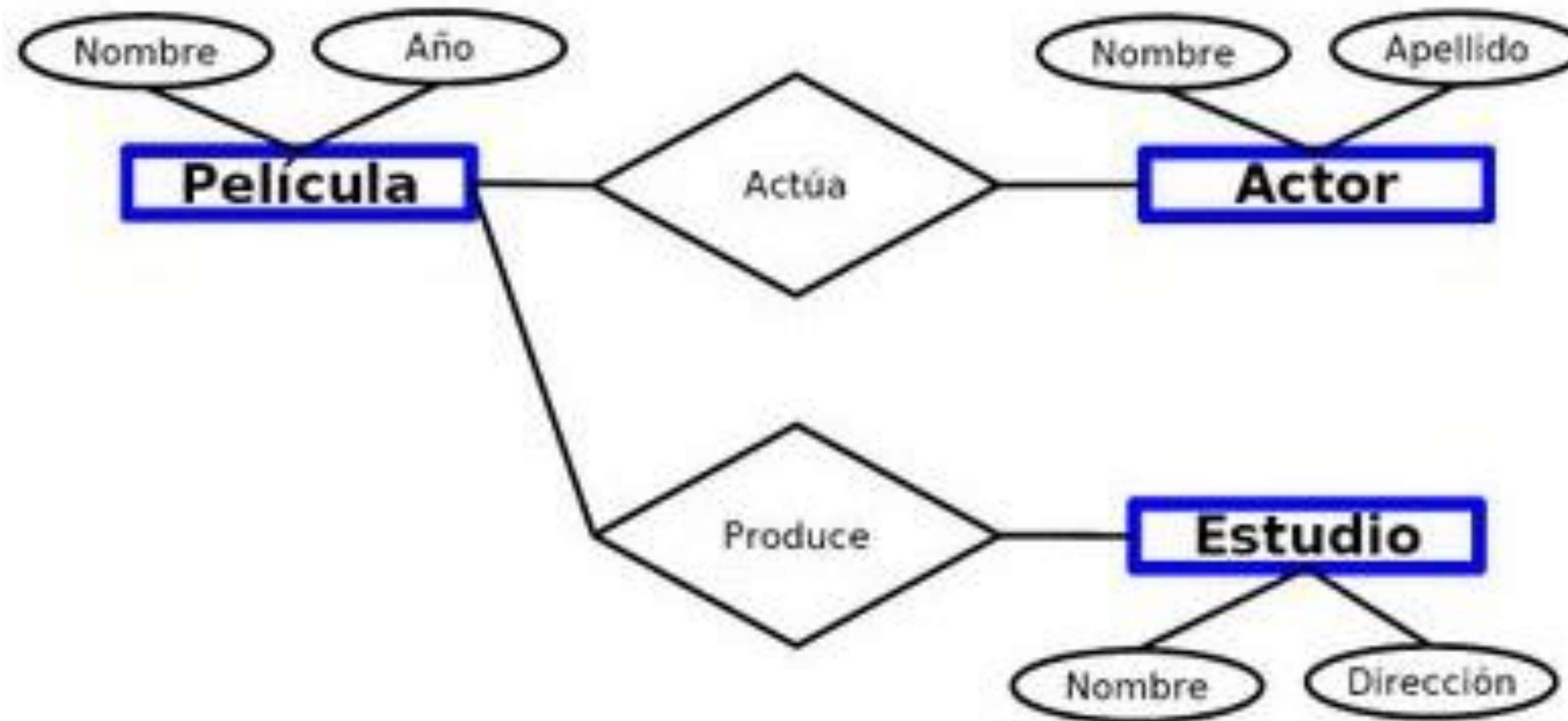
1. Introducción al Modelo Relacional
2. Elementos del Modelo Relacional
3. Notaciones
4. Ejercicios Prácticos

Introducción al Modelo Relacional

- Ventajas del modelo relacional:
- Estructura organizada: Los datos se almacenan en tablas bien definidas, lo que facilita su comprensión y manipulación.
- Flexibilidad: Permite realizar consultas complejas y operaciones de manera eficiente.
- Integridad de los datos: Proporciona mecanismos para garantizar la integridad y consistencia de los datos almacenados.
- Independencia de los datos y las aplicaciones: Permite cambiar la estructura de la base de datos sin afectar a las aplicaciones que acceden a ella.

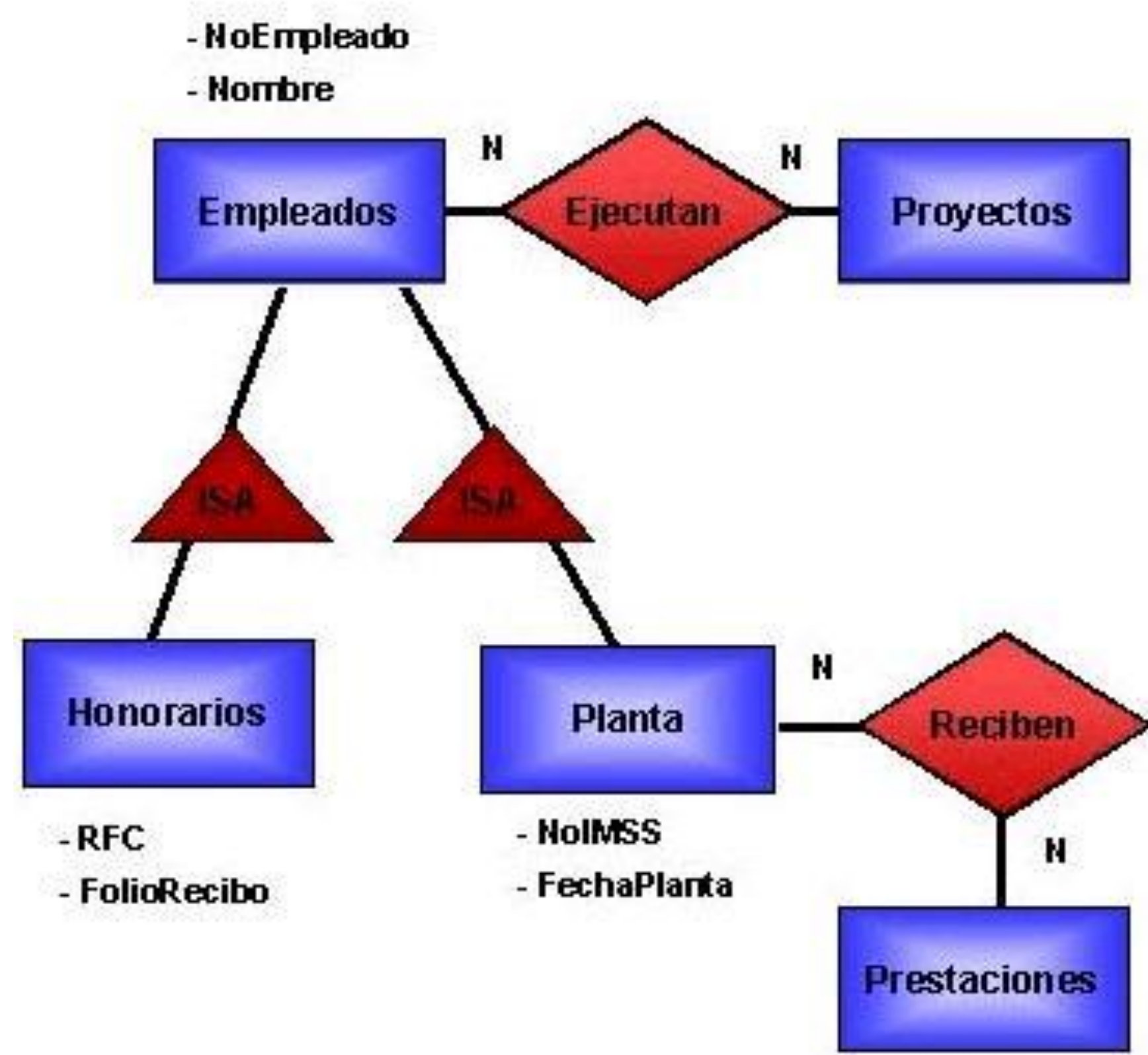


Elementos del Modelo Relacional

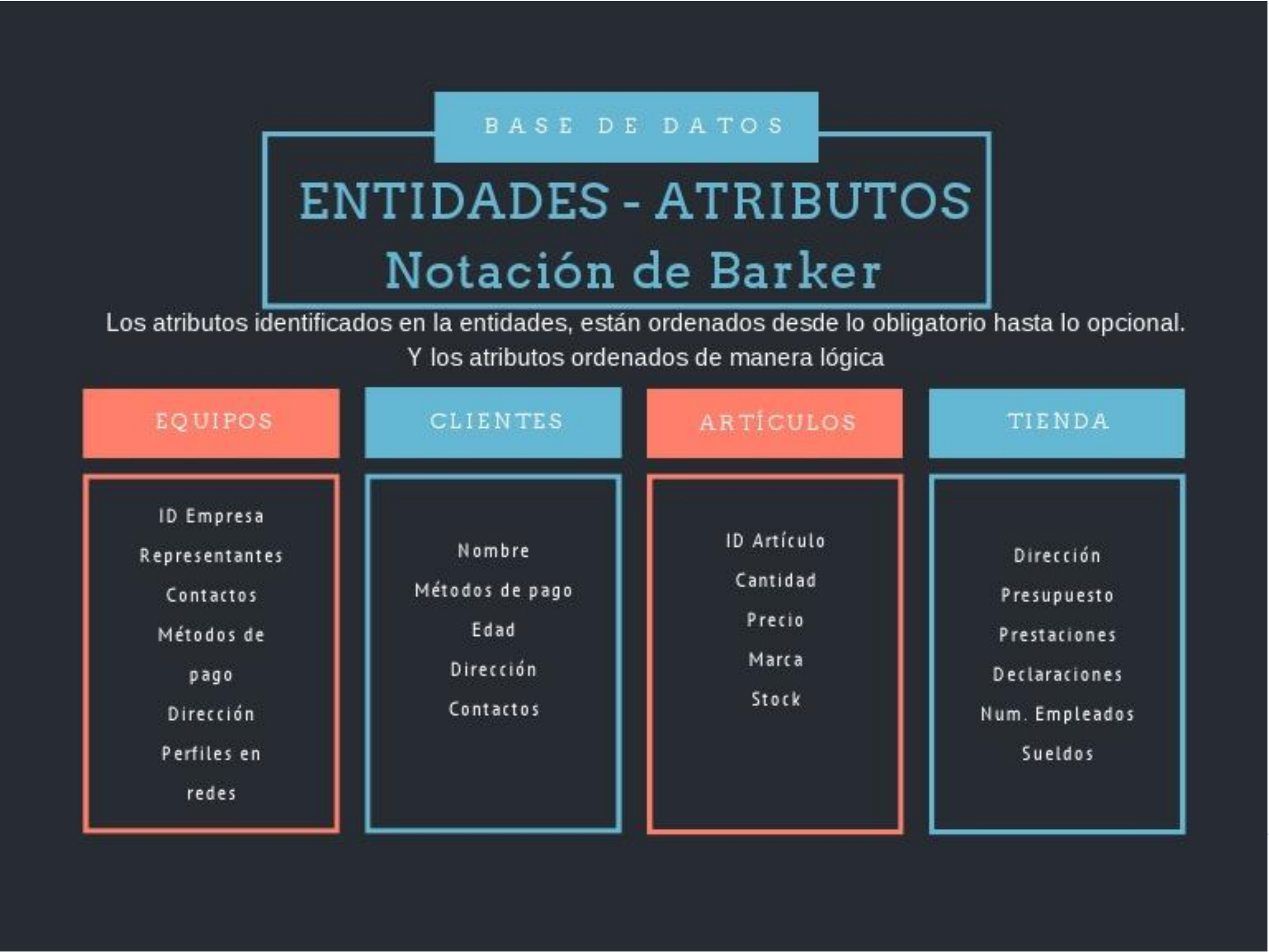


Notaciones

Notaciones CHEN



Notaciones Barker



Ejercicios de modelado 20

Ejercicio: Sistema de Gestión de Empleados

Diseña un modelo de datos para un sistema de gestión de empleados que incluya las siguientes entidades:

Empleados: con atributos de ID, Nombre, Apellido y Departamento_ID.

Departamentos: con atributos de ID y Nombre.
Considera que cada empleado pertenece a un departamento.

SQL

– Creación de la base de datos
CREATE DATABASE EmpresaDB;

USE EmpresaDB;

– Tabla de Empleados
**CREATE TABLE Empleados (
 ID INT PRIMARY KEY,
 Nombre VARCHAR(100),
 Apellido VARCHAR(100),
 Departamento_ID INT
);**

– Inserción de datos de ejemplo en la tabla Empleados
**INSERT INTO Empleados (ID, Nombre, Apellido, Departamento_ID)
VALUES**

**(1, 'Juan', 'Pérez', 1),
(2, 'María', 'Gómez', 2),
(3, 'Pedro', 'López', 1),
(4, 'Ana', 'Martínez', 3),
(5, 'Luis', 'Rodríguez', 2),
(6, 'Elena', 'Sánchez', 1),
(7, 'Carlos', 'García', 3),
(8, 'Laura', 'Hernández', 2),
(9, 'David', 'Díaz', 1),
(10, 'Sofía', 'Ruiz', 3);**

– Tabla de Departamentos
**CREATE TABLE Departamentos (
 ID INT PRIMARY KEY,
 Nombre VARCHAR(100)
);**

– Inserción de datos de ejemplo en la tabla Departamentos
INSERT INTO Departamentos (ID, Nombre) VALUES
**(1, 'Ventas'),
(2, 'Recursos Humanos'),
(3, 'Finanzas'),
(4, 'Tecnología');**

Ejercicios

1. Consulta básica de empleados
2. Consulta de empleados con su respectivo departamento
3. Consulta de empleados del departamento de Ventas
4. Consulta de empleados y sus departamentos (incluyendo los que no tienen departamento asignado)
5. Consulta de empleados que pertenecen a un departamento y sus departamentos
6. Consulta de departamentos con el número de empleados que tienen
7. Consulta de empleados cuyos apellidos comienzan con la letra 'G'
8. Consulta de empleados ordenados alfabéticamente por apellido
9. Consulta de empleados cuyo nombre es 'María' y que pertenecen al departamento de Recursos Humanos
10. Consulta de empleados que no tienen asignado un departamento

Ejercicios de modelado 21

Ejercicio: Sistema de Gestión de una Tienda Online

Diseña un modelo de base de datos para un sistema de gestión de una tienda online que venda productos electrónicos. El modelo debe incluir las siguientes entidades y relaciones:

Clientes: Deben tener un ID, nombre, apellido, correo electrónico y dirección.

Productos: Deben tener un ID, nombre, descripción, precio y cantidad en stock.

Categorías: Los productos deben estar asociados a una categoría. Cada categoría debe tener un ID y un nombre.

Pedidos: Deben tener un ID, fecha, cliente_ID y estado del pedido.

Detalles de Pedidos: Cada pedido debe tener uno o más detalles de pedido asociados que incluyan el ID del pedido, ID del producto, cantidad y precio unitario.

Asegúrate de definir las claves primarias y foráneas apropiadas, así como las relaciones entre las entidades. También considera la normalización de la base de datos para reducir la redundancia y mejorar la integridad de los datos.

SQL

– Creación de la base de datos

```
CREATE DATABASE TiendaOnlineDB;
```

```
USE TiendaOnlineDB;
```

– Tabla de Clientes

```
CREATE TABLE Clientes (  
  ID INT PRIMARY KEY,  
  Nombre VARCHAR(100),  
  Apellido VARCHAR(100),  
  CorreoElectronico VARCHAR(100),  
  Direccion VARCHAR(255)  
);
```

– Tabla de Categorías

```
CREATE TABLE Categorias (  
  ID INT PRIMARY KEY,  
  Nombre VARCHAR(100)  
);
```

– Tabla de Productos

```
CREATE TABLE Productos (  
  ID INT PRIMARY KEY,  
  Nombre VARCHAR(100),  
  Descripcion TEXT,  
  Precio DECIMAL(10, 2),  
  Stock INT,  
  Categoria_ID INT,  
  FOREIGN KEY (Categoria_ID) REFERENCES Categorias(ID)  
);
```

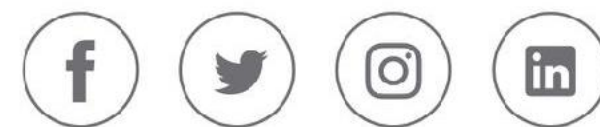
– Tabla de Pedidos

```
CREATE TABLE Pedidos (  
  ID INT PRIMARY KEY,  
  Fecha DATE,  
  Cliente_ID INT,  
  EstadoPedido VARCHAR(50),  
  FOREIGN KEY (Cliente_ID) REFERENCES Clientes(ID)  
);
```

– Tabla de Detalles de Pedidos

```
CREATE TABLE DetallesPedidos (  
  Pedido_ID INT,  
  Producto_ID INT,  
  Cantidad INT,  
  PrecioUnitario DECIMAL(10, 2),  
  PRIMARY KEY (Pedido_ID, Producto_ID),  
  FOREIGN KEY (Pedido_ID) REFERENCES Pedidos(ID),  
  FOREIGN KEY (Producto_ID) REFERENCES Productos(ID)  
);
```

MUCHAS GRACIAS!



inacap.cl