

The logo for CODERHOUSE, featuring the word "CODERHOUSE" in a bold, italicized, black sans-serif font. The text is centered within a light gray rectangular background.

Nombre del Proyecto: Base de Datos para Librería

Nombre y Apellido: Lucila Nemesis Torres

Fecha: 07/10/2025

Comisión: 81840

Introduccion:

Este proyecto consiste en el diseño y en la creación de una base de datos para una librería online. El sistema permite gestionar libros, autores, clientes y ventas de forma eficiente.

Objetivo:

El objetivo de esta base de datos es lograr centralizar y organizar la información de la librería, facilitando la administración de ventas, el control de inventario y seguimiento de los clientes.

Situación problemática:

Actualmente la información de la librería se maneja de manera manual y dispersa, lo cual genera errores, dificulta el acceso rápido a datos y dificulta el análisis de ventas e inventario. Una base de datos relacionales resuelve estos problemas centralizando la información y asegurando que se encuentre siempre actualizada.

Modelo de negocio:

Nuestra librería online vende libros de distintos autores a diferentes clientes. Se necesita registrar cada venta, mantener actualizado el stock de los libros y conocer información tanto de los autores como de los compradores

Listado de tablas y campos:

Tabla: autores

id_autor (INT, PK, AUTO_INCREMENT): identificador único del autor

nombre (VARCHAR(80)): nombre del autor

Tabla: libros

id_libro (INT, PK, AUTO_INCREMENT): identificador único del libro

título (VARCHAR(100)): título del libro

id_autor (INT, FK): referencia al autor

Tabla: clientes

id_cliente (INT, PK, AUTO_INCREMENT): identificador único del cliente

nombre (VARCHAR(100)): nombre del cliente

Tabla: ventas

id_venta (INT, PK, AUTO_INCREMENT): identificador único de la venta

id_cliente (INT, FK): referencia al cliente que compra

fecha (DATE): fecha de la venta

Tabla: detalle_ventas

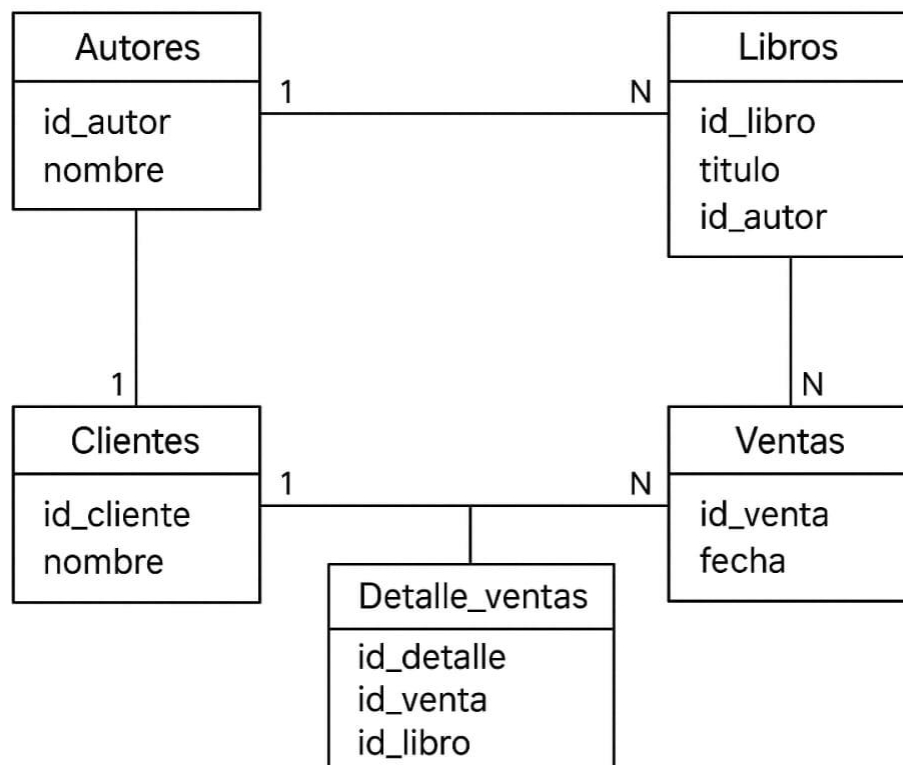
id_detalle (INT, PK, AUTO_INCREMENT): identificador único del detalle

id_venta (INT, FK): referencia a la venta

id_libro (INT, FK): referencia al libro vendido

cantidad (SMALLINT UNSIGNED): cantidad de libros vendidos

precio_unitario (DECIMAL(8,2)): precio de cada libro



LISTADO DE TABLAS Y CAMPOS

Tabla: autores

Abreviatura	Nombre completo del campo	Tipo de dato	Tipo de clave	Descripción
Id_autor	ID del autor	INT	PK, AUTO_INCREMENT	Identificador único del autor
nombre	Nombre del autor	VARCHAR(80)		Nombre del autor

Tabla: libros

Abreviatura	Nombre completo del campo	Tipo de dato	Tipo de clave	Descripción
id_libro	ID del libro	INT	PK, AUTO_INCREMENT	Identificador único del libro
Id_autor	ID del autor	INT	FK	Referencia al autor

Tabla: clientes

Abreviatura	Nombre completo del campo	Tipo de dato	Tipo de clave	Descripción
Id_venta	ID de la venta	INT	PK, AUTO_INCREMENT	Identificador único de la venta
Id_cliente	ID del cliente que compra	INT	FK	Referencia al cliente
fecha	Fecha de la venta	DATE		Fecha de la venta

Tabla: detalle_ventas

Abreviatura	Nombre completo del campo	Tipo de dato	Tipo de clave	Descripción
Id_detalle	ID del detalle	INT	PK, AUTO_INCREMENT	Identificador único del detalle
Id_venta	ID de la venta	INT	FK	Referencia a la venta

id_libro	ID del libro	INT	FK	Referencia al libro vendido
Cantidad	Cantidad de libros vendidos	SMALLINT UNSIGNED		Cantidad vendida
Precio_unitario	Precio unitario del libro	DECIMAL(8,2)		Precio de venta por libro

PK: Primary Key (clave primaria)

FK: Foreign Key (clave foránea)

AUTO_INCREMENT: valor numérico generado automáticamente por el sistema

TRABAJO Nº 2

Listado y descripción de objetos de Base de Datos: Vistas, Función, Procedimiento y Trigger

VISTA: vw_historial_compras_clientes

Descripción: Esta vista permite visualizar el historial de compras de cada cliente. Muestra, para cada venta realizada, el nombre del cliente, la fecha de compra, la cantidad total de libros adquiridos y el monto total gastado.

Objetivo: Facilita la consulta y el análisis de las compras de los clientes, permitiendo generar reportes completos de ventas y conocer el comportamiento de compra para la toma de decisiones comerciales.

Tablas involucradas:

- Clientes
- Ventas
- detalle_ventas

FUNCIÓN: fn_nombre_autor_por_libro

Descripción: La función fn_nombre_autor_por_libro retorna el nombre del autor correspondiente a un libro a partir del ID de dicho libro.

Objetivo: Permite obtener de forma sencilla el autor de un libro dado, útil para consultas, reportes y procesos que requieren mostrar información completa de los libros vendidos o disponibles.

Tablas involucradas:

- autores
- libros

STORED PROCEDURE: sp_insertar_venta

Descripción: El procedimiento sp_insertar_venta automatiza la inserción de una nueva venta: recibe el ID del cliente y la fecha de venta, agrega el registro en la tabla ventas y devuelve el ID generado para poder asociar posteriormente los detalles correspondientes.

Objetivo: Facilita y estandariza el proceso de registro de ventas, reduciendo errores y agilizando la carga de datos, especialmente útil para integraciones futuras con sistemas web o aplicaciones.

Tablas involucradas:

- ventas

TRIGGER: trg_actualizar_stock

Descripción: El trigger trg_actualizar_stock se ejecuta automáticamente cada vez que se ingresa un nuevo detalle de venta. Su función es descontar del stock el número de unidades vendidas de dicho libro en la tabla libros.

Objetivo: Automatiza el control de inventario, asegurando que el stock de cada libro esté siempre actualizado luego de cada venta, y evitando así la sobreventa y los errores de manejo manual.

Tablas involucradas:

- detalle_ventas
- libros