

Trabajo práctico curso SQL

Coderhouse

Alumno: Julia Elizondo

Índice

Índice	1
Temática de la base de datos	2
Diagrama Entidad-Relación	3
Tablas que comprenden la base de datos	4
Vistas	6
Funciones	6
Procedimientos almacenados	6
Triggers	7

Temática de la base de datos

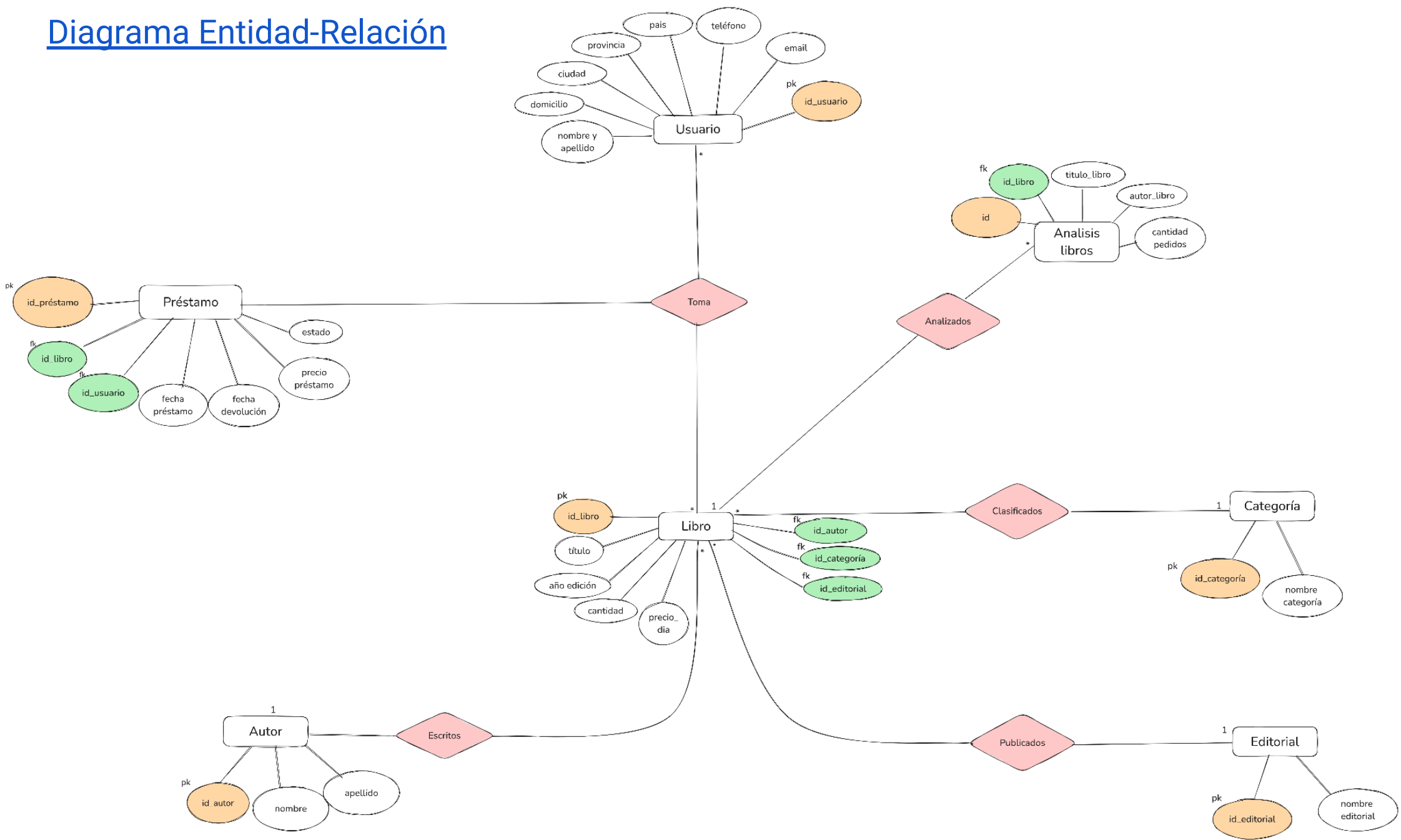
El proyecto se basa en un sistema de gestión para una plataforma de alquiler o préstamo de libros de una biblioteca virtual. El objetivo es poder llevar un registro de los libros con los que cuenta la biblioteca y los usuarios que utilizan sus servicios, así también como poder controlar los préstamos otorgados por la misma.

La plataforma funciona de la siguiente manera: los usuarios eligen uno o más libros que tomarán prestados, pagando un precio determinado por el tiempo de duración del préstamo (en días) y devolviendo el libro una vez finalizado dicho término.

La base de datos almacenará toda la información relativa a los usuarios que interactúan en la plataforma, un registro de los libros que se encuentran disponibles para alquilar, de los autores de los mismos, así también como las categorías en las que se clasifican y las editoriales que los publican.

Además, también contará con información respecto a los préstamos que son tomados por los usuarios, para así poder llevar un control de qué libros están en manos de los usuarios y cuáles aún no han sido prestados y en qué fechas fueron retirados o devueltos.

Diagrama Entidad-Relación



Tablas que comprenden la base de datos

Nombre tabla	Descripción	Campos	Abreviatura campo	Tipo de dato	NULL	Default	Tipo de clave
Autor	Almacena datos de los autores de los libros disponibles en la plataforma	ID autor	id_autor	INT	NOT NULL	AUTO_INCREMENT	PK
		Nombre	nombre	VARCHAR(80)	NOT NULL		
		Apellido	apellido	VARCHAR(80)	NOT NULL		
Editorial	Contiene las editoriales de los libros disponibles en la plataforma	ID editorial	id_editorial	INT	NOT NULL	AUTO_INCREMENT	PK
		Nombre editorial	nombre_editorial	VARCHAR(30)	NOT NULL		
Categoría	Contiene las categorías relativas a los libros de la plataforma.	ID categoría	id_categoria	INT		AUTO_INCREMENT	PK
		Nombre categoría	nombre_categoria	VARCHAR(50)	NOT NULL		
Usuario	Almacena los datos de todos los usuarios registrados en la plataforma	ID usuario	id_usuario	INT	NOT NULL	AUTO_INCREMENT	PK
		Nombre	nombre	VARCHAR(80)	NOT NULL		
		Apellido	apellido	VARCHAR(80)	NOT NULL		
		Fecha nacimiento	fecha_nacimiento	DATE	NOT NULL		
		Domicilio	domicilio	VARCHAR(50)	NOT NULL		
		Ciudad	ciudad	VARCHAR(25)	NOT NULL		
		Provincia	provincia	VARCHAR(25)	NOT NULL		
		País	pais	VARCHAR(25)	NOT NULL		
		Teléfono	teléfono	VARCHAR(15)			
		E-mail	email	VARCHAR(50)	NOT NULL		

Libro	Contiene la información relativa a los libros que están disponibles en la plataforma, tal como nombre, autor, editorial, precio del alquiler de cada libro, entre otros.	ID libro	id_libro	INT	NOT NULL	AUTO_INCREMENT	PK
		ID autor	id_autor	INT			FK
		ID categoría	id_categoria	INT			FK
		ID editorial	id_editorial	INT			FK
		Título	titulo	VARCHAR(100)	NOT NULL		
		Año edición	ano_edicion	YEAR			
		Cantidad	cantidad	INT	NOT NULL		
		Precio_día	precio_dia	FLOAT	NOT NULL		
Préstamo	Contiene los datos de los usuarios que tomaron un libro en préstamo, tomando información de las tablas Libros y Usuarios.	ID préstamo	id_prestamo	INT	NOT NULL	AUTO_INCREMENT	PK
		ID usuario	id_usuario	INT			FK
		ID libro	id_libro	INT			FK
		Fecha préstamo	fecha_prestamo	DATE			
		Fecha devolución	fecha_devolucion	DATE			
		Estado	estado	BOOLEAN			
		Precio_prestamo	precio_prestamo	INT	NOT NULL		
Análisis libros	Tabla creada para contar la cantidad de veces que un libro es solicitado a préstamo, para poder analizar cuáles son los libros más pedidos.	ID	id	INT	NOT NULL	AUTO_INCREMENT	PK
		ID libro	id_libro	INT			FK
		Título libro	titulo_libro	VARCHAR(100)	NOT NULL		
		Autor libro	autor_libro	VARCHAR(160)	NOT NULL		
		Cantidad pedidos	cantidad_pedidos	INT	NOT NULL		

Vistas

La base de datos cuenta con las siguientes vistas:

- **Vw_libros_prestados:** El objetivo de esta vista es mostrar la totalidad de los libros que se encuentran alquilados al momento de la consulta, visualizando datos como id libro, el nombre y apellido del autor, la editorial que publicó el libro, el título, el usuario que lo tomó prestado, la fecha de préstamo y fecha de devolución. Para eso, consulta las tablas Libro y Préstamo.
- **Vw_libros_disponibles:** Esta vista, de forma opuesta a la anterior, muestra los libros que se encuentran disponibles, es decir, que no están prestados al momento. Para su creación también se consultan las tablas Libro y Préstamo, pero en este caso toma aquellos libros cuyo estado de préstamo es False.
- **Vw_usuarios_prestamos_activos:** El objetivo de esta vista es poder consultar rápidamente los datos de los usuarios con préstamos activos. Muestra el id usuario, nombre, apellido, domicilio, ciudad, provincia, email y id préstamo. Para crearla consultamos las tablas Usuario y Préstamo.

Funciones

- **Calcular_prestamo:** se creó esta función para poder obtener de manera rápida el valor de un préstamo determinado, teniendo en cuenta la fecha de inicio, la fecha de devolución, y el valor diario del mismo.
- **Calcular_ingresos_totales_a_fecha:** esta función permite calcular el total de ingresos obtenidos por los préstamos otorgados a una fecha de corte determinada. Para ello, sólo tiene en cuenta los préstamos cuya fecha de inicio es anterior a la fecha de corte.
- **Titulo_libro:** permite obtener de manera rápida el título de un libro determinado, consultando únicamente por el id del mismo.

Procedimientos almacenados

- **sp_insertar_libro:** permite insertar datos dentro de la tabla Libro de manera rápida y efectiva.
- **sp_ordenar_tablas:** permite ordenar las diferentes tablas de la base de datos según un campo determinado. Recibe como parámetro la tabla a ordenar, el campo según el cual se realizará el order by, y por último si el orden se hará de forma ascendente o descendente.
- **sp_devolver_libro:** actualiza la tabla Préstamo para establecer el estado del mismo en "False" cuando un libro es devuelto. Recibe como parámetro el id del préstamo, del libro y del usuario.
- **sp_obtener_libros_por_autor:** este procedimiento muestra los libros existentes escritos por un autor determinado. Muestra tanto id libro, el título, el nombre y apellido del autor, y la editorial.

Triggers

- `verificar_stock`: trigger creado para verificar que un libro tiene stock antes de generar el préstamo, en caso de no tener, lanza un error.
- `disminuir_stock`: creado para actualizar el stock de los libros, disminuyendo el mismo al momento en que se crea un préstamo.
- `aumentar_stock`: trigger creado para actualizar el stock de los libros, aumentando el mismo al momento en que es devuelto y el préstamo pasa a tener estado False.
- `actualizar_cantidad_pedida`: trigger creado con el fin de actualizar la tabla `analisis_libros` , utilizada para llevar un conteo de los libros y así poder hacer un análisis de los más solicitados.