

# Biblioteca Pública N° 175

## Provincia de Mendoza

### **Descripción del proyecto:**

Este proyecto tiene como objetivo principal desarrollar una base de datos para una biblioteca con el fin de gestionar de manera eficiente y organizada los recursos bibliográficos, los usuarios y las transacciones relacionadas. Esta base de datos permitirá agilizar las tareas de administración y facilitará el acceso a la información necesaria para el funcionamiento diario de la biblioteca.

La base de datos contiene tablas principales para representar entidades clave, como libros, autores, usuarios, préstamos y devoluciones. Cada tabla estará diseñada de manera que refleje la relación y la estructura adecuada de los datos, garantizando la integridad y la coherencia de la información almacenada.

Además, se implementarán funcionalidades como la gestión de inventario, donde se podrán registrar los detalles de cada libro, incluyendo el título, el autor, la categoría y la disponibilidad. También se podrán agregar y actualizar registros de usuarios, manteniendo un seguimiento de los datos personales, las membresías y el historial de préstamos.

La base de datos permitirá realizar consultas eficientes para buscar libros por título, autor o categoría, y también para verificar el estado de préstamos, fechas de devolución y multas. Además, se integrará un sistema de generación de informes y estadísticas que facilitará la toma de decisiones y la planificación de actividades futuras.

Este proyecto busca mejorar la eficiencia en la gestión de la biblioteca, optimizando los procesos de préstamo, devolución, inventario y seguimiento de usuarios. Con esta base de datos, esperamos brindar a los usuarios un acceso más rápido y preciso a los recursos bibliográficos.

### **Introducción:**

Nuestra base de datos es el corazón de este sistema, permitiéndonos organizar y acceder a una extensa colección de libros, novelas, ensayos y otros recursos bibliográficos de manera ágil y precisa. A través de esta solución tecnológica, hemos creado un entorno virtual donde se fusiona lo clásico con lo moderno, proporcionando una experiencia única para nuestros usuarios.

### **Objetivo:**

El objetivo de esta DB es organizar y acceder de manera eficiente a las diferentes colecciones de libros y recursos bibliográficos, pudiendo buscar y localizar materiales fácilmente, también realizar un seguimiento preciso de préstamos y devoluciones y poder generar informes detallados de la biblioteca

### **Situación problemática:**

El anticuado sistema de control y administración de la biblioteca presentaba algunos problemas como por ejemplo:

- Gestión eficiente de la colección
- Agilidad en la búsqueda y recuperación de información
- Control de inventario
- Registro de préstamos y devoluciones
- Generación de informes y estadísticas
- Gestión eficiente de usuarios

### **Modelo de negocio:**

Préstamo de materiales: La biblioteca proporciona servicios de préstamo de libros, manuales, ensayos, y otros recursos bibliográficos a sus miembros. Estos materiales se prestan por un período determinado y pueden estar sujetos a políticas de renovación y multas por retraso o pérdida.

Venta de libros: Además de los servicios de préstamo, la biblioteca puede generar ingresos a través de la venta de libros. Esto puede incluir la venta de libros usados, donados o duplicados de la colección de la biblioteca.

El modelo de negocio de la biblioteca se basa en la financiación a través de fuentes como alquiler, venta, donaciones, patrocinios y eventos de recaudación de fondos. El objetivo principal es brindar acceso gratuito o asequible a la información, promover la alfabetización, el aprendizaje y la participación comunitaria, y enriquecer la vida cultural de la comunidad a la que sirve.

### **Objetos de mantenimiento de la biblioteca:**

1. **Tablas:** Las tablas son los objetos principales de una base de datos. Almacenan y organizan los datos en filas y columnas. Cada tabla representa una entidad o un conjunto de datos relacionados
2. **Índices:** Los índices se utilizan para mejorar la velocidad y eficiencia de las consultas en una base de datos. Aceleran la recuperación de datos al crear estructuras de datos especiales que permiten acceder rápidamente a la información requerida.
3. **Vistas:** Las vistas son consultas almacenadas que se comportan como tablas virtuales. Proporcionan una forma conveniente de acceder a datos complejos o resumidos, ocultando la complejidad de las consultas subyacentes.
4. **Procedimientos almacenados:** Los procedimientos almacenados son bloques de código SQL que se guardan en la base de datos y se pueden ejecutar cuando sea necesario. Ayudan a encapsular la lógica empresarial y las operaciones recurrentes para su reutilización y mantenimiento.
5. **Disparadores (Triggers):** Los disparadores son objetos que se ejecutan automáticamente en respuesta a ciertos eventos o acciones en la base de datos, como la inserción, actualización o eliminación de datos en una tabla.

Se utilizan para aplicar lógica adicional o realizar acciones adicionales cuando se producen ciertos cambios.

6. **Funciones:** Las funciones son bloques de código que se utilizan para realizar cálculos o manipulaciones de datos y devolver un resultado. Pueden ser utilizadas en consultas SQL y en otros objetos de la base de datos.
7. **Restricciones:** Las restricciones son reglas que se aplican a las tablas para garantizar la integridad de los datos. Pueden incluir restricciones de clave primaria, clave foránea, valores únicos, entre otros. Ayudan a mantener la coherencia y la consistencia de los datos en la base de datos.
8. **Todos los objetos de mantenimiento se encuentran comentados en el código**

### **Ejemplos de consultas SQL que permiten generación de informes simples:**

1. Cantidad de unidades vendidas y sumatoria total.

```
1 • select * from ventas;  
2  
3 • SELECT SUM(Cantidad) AS Unidades_vendidas, SUM(Monto_Facturado) AS Total  
4 FROM ventas;  
5
```

	Unidades_vendidas	Total
►	96	287.3800005912781

## 2. Filtro de informes por edad

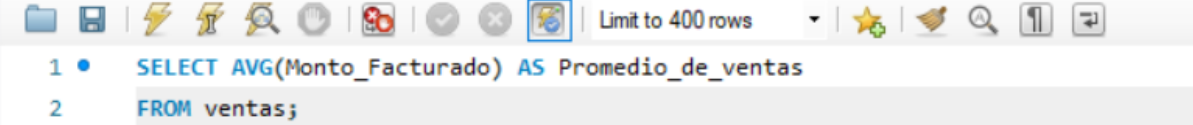
Limit to 400 rows

```
1 • SELECT *
2 FROM usuarios
3 WHERE Edad < 25
4 LIMIT 18;
```

Result Grid

	id_usuario	Nombre	Apellido	Edad	Telefono	Direccion
▶	8	Elena	Sánchez	23	+1 (555) 890-1234	Avenida del Sol #5678, Guadalajara
	36	Pedro	Vázquez	19	+1 (555) 345-6789	Calle 11 #1234, Tijuana
	41	Ana	López	22	+1 (555) 890-1234	Calle 44 #123, Monterrey
	45	María	Martínez	20	+1 (555) 234-5678	Avenida 25 #567, Querétaro
	48	Isabella	Sánchez	22	+1 (555) 567-8901	Calle 88 #6789, Ciudad de México
	51	Pedro	López	19	+1 (555) 901-2345	Avenida 40 #789, Cancún
	56	Benjamin	Schwarz	24	+49 (0) 177 9876543	Hochstraße 7, Düsseldorf
	61	Lisa	Becker	22	+49 (0) 177 1234568	Goethestraße 2, Hannover
	66	David	Kramer	23	+49 (0) 176 2345670	Mühlenstraße 4, Mannheim
	77	Sofía	Martínez	24	+57 1 2345678	Carrera 7 # 71-52, Bogotá
	81	Isabella	Hernan...	22	+57 4 123 4567	Carrera 43A # 7-50, Medellín
	97	Valentina	Perez	24	+57 4 567 8901	Carrera 70 # 45-24, Medellín
	112	Benjamin	Rodríguez	24	+1 (555) 567-8901	789 Elm St, San Diego

### 3. Informe de monto de las ventas promedio de la biblioteca

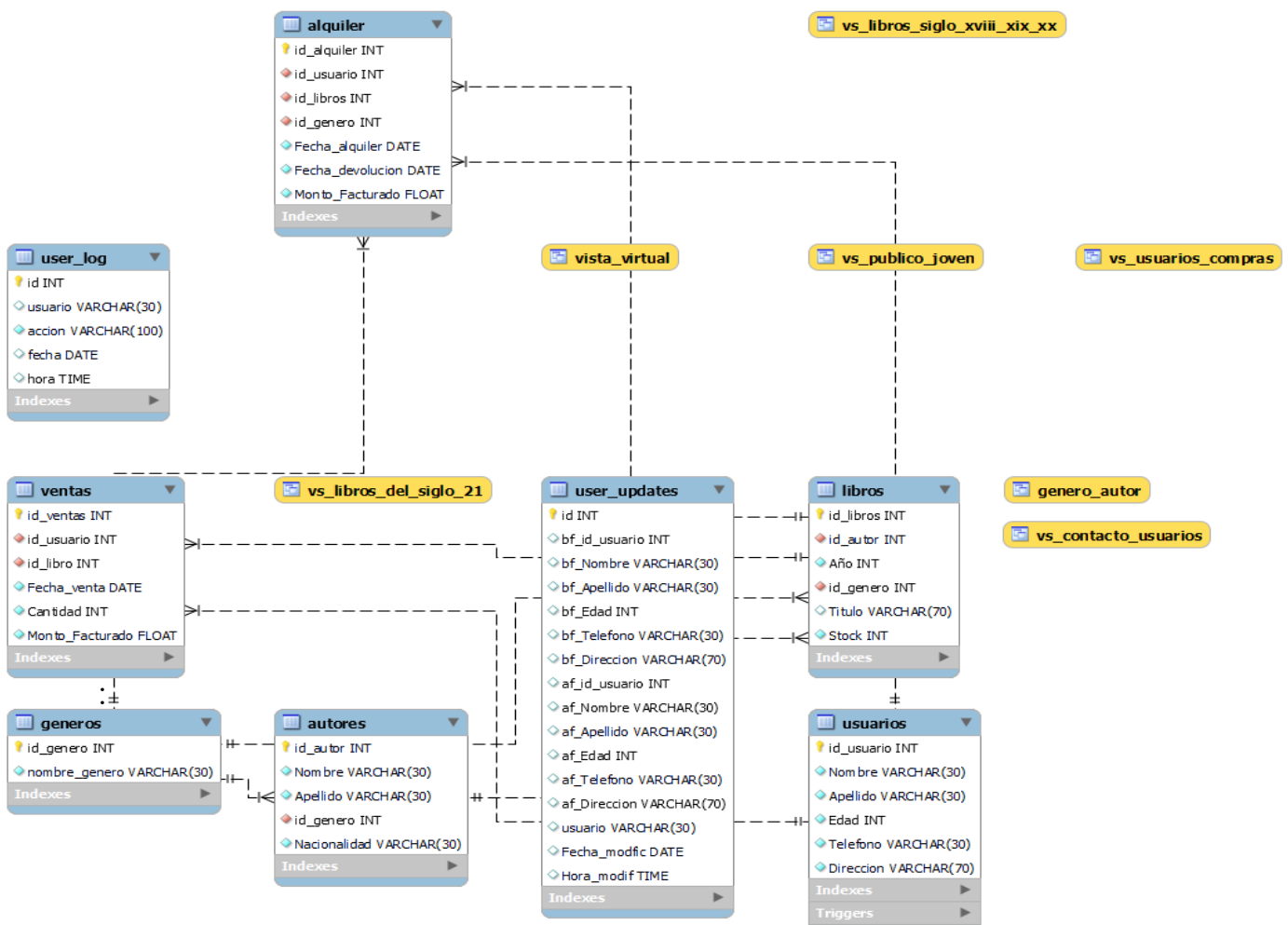


```
1 • SELECT AVG(Monto_Facturado) AS Promedio_de_ventas
2 FROM ventas;
```

Result Grid

Promedio_de_ventas
13.684761932918004

### Diagrama Entidad Relación:



## **Lista de tablas:**

### **1. Alquiler**

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Alquiler	id_alquiler	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Alquiler	id_usuario	INT	NOT NULL	N/A	N/A	FK
Alquiler	id_libros	INT	NOT NULL	N/A	N/A	FK
Alquiler	id_genero	INT	NOT NULL	N/A	N/A	FK
Alquiler	Fecha_alquiler	DATE	NOT NULL	N/A	N/A	N/A
Alquiler	Fecha_devolucion	DATE	NOT NULL	N/A	N/A	N/A
Alquiler	Monto_Facturado	FLOAT	NOT NULL	N/A	N/A	N/A

### **2. Autores**

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Autores	id_autor	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Autores	Nombre	VARCHAR (30)	NOT NULL	N/A	N/A	N/A
Autores	Apellido	VARCHAR (30)	NOT NULL	N/A	N/A	N/A
Autores	id_genero	INT	NOT NULL	N/A	N/A	FK
Autores	Nacionalidad	VARCHAR (30)	NOT NULL	N/A	N/A	N/A

### **3. Géneros**

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Generos	id_genero	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Generos	nombre_genero	VARCHAR (30)	NOT NULL	N/A	N/A	N/A

#### 4. Libros

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Libros	id_libros	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Libros	id_autor	INT	NOT NULL	N/A	N/A	FK
Libros	Año	INT	NOT NULL	N/A	N/A	N/A
Libros	id_genero	INT	NOT NULL	N/A	N/A	FK
Libros	Titulo	VARCHAR (70)	NOT NULL	N/A	N/A	N/A
Libros	Stock	INT	NOT NULL	N/A	N/A	N/A

#### 5. Usuarios

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Usuarios	id_usuario	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Usuarios	Nombre	VARCHAR (30)	NOT NULL	N/A	N/A	N/A
Usuarios	Apellido	VARCHAR (30)	NOT NULL	N/A	N/A	N/A
Usuarios	Edad	INT	NOT NULL	N/A	N/A	N/A
Usuarios	Telefono	VARCHAR (30)	NOT NULL	N/A	N/A	N/A
Usuarios	Direccion	VARCHAR (30)	NOT NULL	N/A	N/A	N/A

#### 6. Ventas

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
Ventas	id_ventas	INT	AUTOINCREMENT	NOT NULL	N/A	PK
Ventas	id_usuario	INT	NOT NULL	N/A	N/A	FK
Ventas	id_libro	INT	NOT NULL	N/A	N/A	FK
Ventas	Fecha_venta	DATE	NOT NULL	N/A	N/A	N/A
Ventas	Cantidad	INT	NOT NULL	N/A	N/A	N/A
Ventas	Monto_Facturado	FLOAT	NOT NULL	N/A	N/A	N/A



## 7. User\_logs

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
user_log	ID	INT	AUTOINCREMENT	N/A	N/A	PK
user_log	Usuario	VARCHAR (30)	N/A	N/A	N/A	N/A
user_log	Accion	VARCHAR (255)	N/A	N/A	N/A	N/A
user_log	Fecha	DATE	N/A	N/A	N/A	N/A
user_log	Hora	TIME	N/A	N/A	N/A	N/A

## 8. User\_update

Tabla:	Columnas:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de dato:	Tipo de clave:
user_update	id	INT	AUTOINCREMENT	N/A	N/A	PK
user_update	bf_id_usuario	INT	N/A	N/A	N/A	N/A
user_update	bf_Nombre	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	bf_Apellido	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	bf_Edad	INT	N/A	N/A	N/A	N/A
user_update	bf_Telefono	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	bf_Direccion	VARCHAR (70)	N/A	N/A	N/A	N/A
user_update	af_id_usuario	INT	N/A	N/A	N/A	N/A
user_update	af_Nombre	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	af_Apellido	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	af_Edad	INT	N/A	N/A	N/A	N/A
user_update	af_Telefono	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	af_Direccion	VARCHAR (70)	N/A	N/A	N/A	N/A
user_update	usuario	VARCHAR (30)	N/A	N/A	N/A	N/A
user_update	Fecha_modific	DATE	N/A	N/A	N/A	N/A
user_update	Hora_modif	TIME	N/A	N/A	N/A	N/A

## **Descripción Tablas:**

### **1. Alquiler:**

Tabla "Alquiler": En esta tabla se registran los detalles de los libros que han sido alquilados por los usuarios de la biblioteca. Puedes encontrar información como fecha de alquiler y devolución, esta tabla está pensada para gestionar todos los alquileres eficientemente.

### **2. Autores:**

Tabla "Autores": Aquí se guardan los datos de los autores cuyas obras se encuentran en la biblioteca. Puedes encontrar información como el nombre completo del autor, su nacionalidad y más información acerca de los autores.

### **3. Géneros:**

Tabla "Géneros": Esta tabla almacena los diversos géneros literarios presentes en tu biblioteca. Desde acción hasta teatro, romance con un total de 20 géneros.

### **4. Libros:**

Tabla "Libros": Aquí encontrarás la información detallada de cada libro en tu biblioteca. Desde los clásicos hasta las últimas novedades, también puedes encontrar detalles como el título, el autor, el año de publicación y stock disponible.

### **5. Usuarios:**

Tabla "Usuarios": Esta tabla contiene la información de las personas que son parte de la comunidad de tu biblioteca. Encontrarás detalles como el nombre, la dirección y la información de contacto de cada usuario.

### **6. Ventas:**

Tabla "Ventas": Aquí se registran las transacciones de venta de libros realizadas en la biblioteca. Puedes encontrar información como el id del libro y id del autor del libro vendido, la fecha de la venta, el precio de venta y la cantidad.

## **BITÁCORA**

### **7. User\_log:**

La función de esta tabla es registrar cada vez que algún usuario de la biblioteca es agregado o eliminado de la base de datos.

### **8. User\_Update:**

La función de esta tabla es registrar cada vez que modifica la información de algún usuario, teniendo la posibilidad de observar la información anterior a la modificación.

### **Listado de vistas:**

#### **1. vs\_libros\_del\_siglo\_21**

Esta vista está compuesta por las columnas Año, id\_genero, Título y Stock, el objetivo es facilitar la búsqueda de libros modernos del año 2001 en adelante.

#### **2. vs\_libros\_Siglo\_XVIII\_XIX\_XX**

Esta vista está compuesta por las columnas id\_libro, título y año, con el objetivo de facilitar la búsqueda de libros contemporáneos y antiguos desde el año 1701 hasta el año 2000.

#### **3. vs\_contacto\_usuarios**

Esta vista está compuesta por las columnas id\_usuario, Nombre, Apellido y Teléfono, el objetivo es permitir al usuario de la base de datos ponerse rápidamente en contacto a través del teléfono de cualquier usuario y sus respectivos nombres

#### **4. vs\_usuarios\_compras**

Esta vista está compuesta por las columnas id\_usuario, Cantidad y Monto facturado, el objetivo de esta vista es filtrar cuales son los usuarios que están realizando compras para poder ofrecerles descuentos en otros libros

#### **5. vs\_publico\_joven**

Esta vista está compuesta por las columnas id\_usuario, Nombre, Apellido y Edad, el objetivo de esta vista es filtrar la edad de los usuarios para identificar el público joven de la biblioteca y poder hacer eventos y promociones para un público objetivo y fomentar la lectura.

### **Listado de funciones:**

#### **1. Función: func\_caja\_registradora**

Esta función está creada con el objetivo de facilitar los cálculos matemáticos de las ventas realizadas, cumpliendo con el cálculo matemático de multiplicación siendo así la cantidad de artículos \* \$ precio del articulo/s

#### **2. Función: func\_contacto\_usuarios**

Esta función está creada con el objetivo de poder contactar a cualquier usuario ya sea por una promoción o por demora en la devolución de un libro, permitiendo ingresar el id de cualquier usuario y retornando su contacto

### **Listado de S.P:**

#### **1. sp\_libros\_biblioteca**

En este caso el SP está pensado para que el bibliotecario tenga acceso a todos los libros de la biblioteca con solo un botón en el sistema de gestión. interactúa con tabla libros

#### **2. sp\_modify\_user**

En este caso el SP está pensado para que pueda borrar un usuario existente y modificar el id en caso de estar mal cargado en la tabla. interactúa con tabla usuarios

### **3. sp\_order\_libros**

En este caso el SP está pensado para hacer un filtro de los nuevos libros que tenemos en la biblioteca, esto facilita ver los libros y autores más recientes. interactúa con tabla libros

### **4. sp\_libros\_por\_autor**

En este caso el SP está pensado para ingresar un dato de entrada (Autor) y nos regresa los libros y géneros que pertenecen al mismo. interactúa con tabla autores

## **Herramientas, Tecnologías y desarrollo del proyecto**

En este proyecto, se implementó un sistema de gestión de bases de datos relacionales (RDBMS) para administrar y gestionar eficientemente la base de datos utilizada. El RDBMS es un software especializado que proporciona funcionalidades para el almacenamiento, organización y manipulación de los datos.

En particular, se utilizó el lenguaje estándar SQL (Structured Query Language) como medio de interacción con la base de datos. SQL es un lenguaje de programación diseñado específicamente para trabajar con bases de datos relacionales. Permite realizar diversas operaciones, como la creación de estructuras de datos, inserción y modificación de registros, consultas de información y gestión de permisos.

Para implementar la base de datos en este proyecto, se eligió MySQL como el sistema de gestión de bases de datos. MySQL proporciona una amplia gama de características, incluyendo soporte para transacciones, alta disponibilidad y escalabilidad, y herramientas de administración y desarrollo.

### **¿Qué se desarrolló en este proyecto?**

- Diagrama Entidad Relación
- Consultas y subconsultas SQL
- Sublenguaje DDL
- Sublenguajes DCL y TCL
- Tipos de Joins
- Vistas
- Workshop DDL, DCL y TCL
- Sublenguaje DML
- Funciones
- Stored procedures
- Triggers
- DCL y TCL