

# Documentación del Proyecto: Biblioteca

## 1. Descripción General del Proyecto

La base de datos `biblioteca` está diseñada para gestionar de manera eficiente los libros, usuarios, préstamos, devoluciones, multas, reservas, sedes y tipos de usuarios de una biblioteca. Su objetivo principal es permitir el control del inventario, el historial de préstamos y la información de los usuarios, garantizando la integridad de los datos y facilitando la administración del sistema.

## 2. Estructura de Tablas (15)

### a) Autores

- **Descripción:** Almacena la información de los autores de los libros.
- **Campos:**
  - `id_autor` : Identificador único del autor.
  - `nombre_autor`, `apellido_autor` : Nombre completo del autor.
  - `nacionalidad` : Nacionalidad del autor.

### b) Usuarios

- **Descripción:** Contiene los datos de los lectores registrados en la biblioteca.
- **Campos:**
  - `id_usuario` : Identificador único del usuario.
  - `nombre`, `apellido` : Nombre completo del usuario.
  - `direccion`, `telefono`, `correo_electronico` : Información de contacto.
  - `id_tipo_usuario` : Relación con el tipo de usuario.

### c) Editoriales

- **Descripción:** Representa las editoriales que publican los libros.
- **Campos:**
  - `id_editorial` : Identificador único de la editorial.
  - `nombre_editorial`, `ciudad_editorial` : Nombre y ubicación de la editorial.

### d) Libros

- **Descripción:** Contiene la información de los libros disponibles en la biblioteca.
- **Campos:**
  - `id_libro` : Identificador único del libro.
  - `titulo`, `fecha_publicacion` : Información del libro.
  - `id_genero`, `id_categoria`, `id_editorial` : Relaciones con géneros, categorías y editoriales.
  - `disponibilidad` : Indica si el libro está disponible para préstamo.

### e) Autores\_Libros

- **Descripción:** Tabla intermedia para gestionar la relación muchos a muchos entre autores y libros.
- **Campos:**
  - `id_autor_libro` : Clave primaria.
  - `id_autor`, `id_libro` : Claves foráneas hacia `Autores` y `Libros`.

### f) Prestamos

- **Descripción:** Registra los préstamos de libros a usuarios.
- **Campos:**
  - `id_prestamo` : Identificador único del préstamo.
  - `fecha_prestamo`, `fecha_devolucion_estimada`, `fecha_devolucion_real` : Fechas de préstamo y devolución.
  - `estado_prestamo` : Estado actual del préstamo (pendiente, devuelto, etc.).
  - `id_usuario`, `id_libro`, `id_sede` : Relaciones con `Usuarios`, `Libros` y `Sedes`.

### g) Devoluciones

- **Descripción:** Registra las devoluciones reales de libros.
- **Campos:**
  - `id_devolucion` : Identificador único de la devolución.
  - `id_prestamo` : Relación con el préstamo.
  - `fecha_devolucion`, `multa_aplicada`, `observaciones`.

### h) Multas

- **Descripción:** Multas aplicadas a usuarios por retraso o daño de libros.
- **Campos:**
  - `id_multa` : Identificador único de la multa.
  - `id_usuario` : Relación con el usuario.
  - `monto, fecha_aplicacion, pagada` .

#### i) Generos

- **Descripción:** Categoría literaria de los libros.
- **Campos:**
  - `id_genero` : Identificador único del género.
  - `nombre_genero` : Nombre del género (Ficción, Novela, etc.).

#### j) Reservas

- **Descripción:** Usuarios pueden reservar libros no disponibles.
- **Campos:**
  - `id_reserva` : Identificador único de la reserva.
  - `id_usuario, id_libro` : Relaciones con `Usuarios` y `Libros` .
  - `fecha_reserva, estado` .

#### k) Sedes

- **Descripción:** Sedes físicas de la biblioteca.
- **Campos:**
  - `id_sede` : Identificador único de la sede.
  - `nombre_sede, direccion, telefono` .

#### l) Ejemplares

- **Descripción:** Copias de libros en sedes específicas.
- **Campos:**
  - `id_ejemplar` : Identificador único del ejemplar.
  - `id_libro, id_sede` : Relaciones con `Libros` y `Sedes` .
  - `ubicacion, estado` .

#### m) TiposUsuario

- **Descripción:** Tipos de usuarios (estudiante, docente, etc.).
- **Campos:**
  - `id_tipo_usuario` : Identificador único del tipo.
  - `nombre_tipo, prestamos_maximos, dias_prestamo` .

#### n) HistorialPrestamos (Tabla de Hechos)

- **Descripción:** Tabla central que almacena métricas y datos de análisis de préstamos.
- **Campos:**
  - `id_registro` : Identificador único del registro.
  - `id_usuario, id_libro, id_prestamo` : Relaciones con dimensiones.
  - `dias_retraso, multa_generada, fecha_inicio, fecha_fin` .

#### o) Categorias

- **Descripción:** Categorías temáticas de libros (Literatura, Ciencia, etc.).
- **Campos:**
  - `id_categoria` : Identificador único de la categoría.
  - `nombre_categoria, descripcion` .

---

### 3. Vistas (5)

#### a) v\_libros\_disponibles

- **Objetivo:** Mostrar los libros que están disponibles para préstamo.
- **Composición:** `Libros, Editoriales, Generos` .
- **Campos mostrados:** `titulo, nombre_editorial, nombre_genero, disponibilidad` .

#### b) v\_prestamos\_activos

- **Objetivo:** Listar los préstamos que aún no han sido devueltos.
- **Composición:** `Prestamos, Usuarios, Libros` .
- **Campos mostrados:** `nombre, apellido, titulo, fecha_prestamo, fecha_devolucion_estimada` .

#### c) `v_historial_prestamos_usuario`

- **Objetivo:** Mostrar el historial de préstamos de un usuario.
- **Composición:** Prestamos , Usuarios , Libros .
- **Campos mostrados:** nombre , apellido , titulo , fecha\_prestamo , fecha\_devolucion\_real , estado\_prestamo .

#### d) `v_autores_libros`

- **Objetivo:** Relacionar autores con los libros que han escrito.
- **Composición:** Autores\_Libros , Autores , Libros .
- **Campos mostrados:** nombre\_autor , apellido\_autor , titulo .

#### e) `v_multas_pendientes`

- **Objetivo:** Mostrar las multas pendientes de pago por usuario.
- **Composición:** Multas , Usuarios .
- **Campos mostrados:** nombre , apellido , monto , fecha\_aplicacion .

---

## 4. Funciones (3)

#### a) `fn_calcular_dias_retraso`

- **Objetivo:** Calcular cuántos días de retraso tiene un préstamo.
- **Entradas:** fecha\_devolucion\_real , fecha\_devolucion\_estimada .
- **Salida:** Número de días de retraso (0 si no hay retraso).
- **Tabla(s) usada(s):** Prestamos .

#### b) `fn_total_prestamos_usuario`

- **Objetivo:** Contar cuántos préstamos ha realizado un usuario.
- **Entradas:** p\_id\_usuario .
- **Salida:** Número total de préstamos.
- **Tabla(s) usada(s):** Prestamos .

#### c) `fn_libros_por_genero`

- **Objetivo:** Contar cuántos libros hay en un género específico.
- **Entradas:** p\_genero .
- **Salida:** Número total de libros en ese género.
- **Tabla(s) usada(s):** Libros , Generos .

---

## 5. Stored Procedures (3)

#### a) `sp_registrar_prestamo`

- **Objetivo:** Registrar un nuevo préstamo y actualizar la disponibilidad del libro.
- **Entradas:** p\_id\_usuario , p\_id\_libro , p\_fecha\_prestamo , p\_fecha\_devolucion\_estimada , p\_id\_sede .
- **Acciones:**
  - Inserta un nuevo registro en Prestamos .
  - Actualiza la disponibilidad del libro a false .
- **Tabla(s) usada(s):** Prestamos , Libros .

#### b) `sp_devolver_libro`

- **Objetivo:** Registrar la devolución de un libro y actualizar su disponibilidad.
- **Entradas:** p\_id\_prestamo , p\_fecha\_devolucion .
- **Acciones:**
  - Actualiza la fecha de devolución y el estado del préstamo.
  - Cambia la disponibilidad del libro a true .
- **Tabla(s) usada(s):** Prestamos , Libros .

#### c) `sp_buscar_libros_por_genero`

- **Objetivo:** Buscar todos los libros de un género específico.
- **Entradas:** p\_genero .
- **Salida:** Resultado de la consulta con título, editorial y género.
- **Tabla(s) usada(s):** Libros , Editoriales , Generos .

## 6. Triggers (3)

---

### a) `tr_actualizar_disponibilidad_libro`

- **Objetivo:** Actualizar automáticamente la disponibilidad de un libro cuando se registra un préstamo.
- **Evento:** AFTER INSERT en `Prestamos`.
- **Acción:** Cambia `disponibilidad` a `false` en la tabla `Libros` para el libro prestado.
- **Tabla(s) afectada(s):** `Libros`.

### b) `tr_actualizar_disponibilidad_devolucion`

- **Objetivo:** Actualizar automáticamente la disponibilidad de un libro cuando se registra su devolución.
- **Evento:** AFTER UPDATE en `Prestamos`.
- **Acción:** Cambia `disponibilidad` a `true` si la devolución se registra.
- **Tabla(s) afectada(s):** `Libros`.

### c) `tr_eliminar_libro_prestamo`

- **Objetivo:** Eliminar préstamos asociados a un libro antes de eliminar el libro.
- **Evento:** BEFORE DELETE en `Libros`.
- **Acción:** Elimina registros en `Prestamos` donde `id_libro` coincide.
- **Tabla(s) afectada(s):** `Prestamos`.

---

## 7. Inserciones de Datos

---

Se han insertado **registros** en cada tabla para probar el funcionamiento del sistema. Se incluyen libros, autores, usuarios, préstamos, devoluciones, multas, reservas, sedes, ejemplares, tipos de usuario, géneros y categorías.

---

## 8. Beneficios del Sistema

---

- **Automatización:** El uso de triggers y stored procedures automatiza procesos como la actualización de disponibilidad.
- **Seguridad y control:** Las vistas permiten acceso controlado a la información sensible.
- **Flexibilidad:** Las funciones permiten cálculos dinámicos como el cálculo de retrasos.
- **Integridad de datos:** Las claves foráneas garantizan la consistencia entre tablas.
- **Análisis:** La tabla de hechos `HistorialPrestamos` permite análisis de métricas como retrasos y multas.