

# Código de SQL

```
CREATE DATABASE biblioteca;  
USE biblioteca;
```

-- Tabla de libros

```
CREATE TABLE libros (  
    id_libro INT PRIMARY KEY AUTO_INCREMENT,  
    titulo VARCHAR(100),  
    autor VARCHAR(100),  
    genero VARCHAR(50),  
    publicacion DATE,  
    cantidad_disponible INT,  
    calificacion INT);
```

-- Tabla de usuarios

```
CREATE TABLE usuarios (  
    id_usuario INT PRIMARY KEY AUTO_INCREMENT,  
    nombre VARCHAR(100),  
    direccion VARCHAR(150),  
    telefono VARCHAR(20));
```

-- Tabla de préstamos

```
CREATE TABLE prestamos (  
    id_prestamo INT PRIMARY KEY AUTO_INCREMENT,  
    id_libro INT,  
    id_usuario INT,  
    fecha_prestamo DATE,  
    fecha_devolucion DATE,  
    devuelto BOOLEAN,  
    FOREIGN KEY (id_libro) REFERENCES libros(id_libro),  
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario));
```

-- Tabla de reportes

```
CREATE TABLE reportes (  
    id_reporte INT PRIMARY KEY AUTO_INCREMENT,  
    tipo_reporte VARCHAR(50),  
    fecha_generacion DATE,  
    descripcion TEXT,  
    resultado TEXT,  
    usuario_genero VARCHAR(50));
```

# Explicación del código SQL

El código comienza creando una base de datos llamada biblioteca y seleccionándola con el comando `USE biblioteca;`. Esto asegura que todas las tablas que se creen a continuación pertenezcan a esa base de datos.

## 1. Tabla libros

La tabla libros almacena la información de cada ejemplar disponible en la biblioteca.

**id\_libro:** es un número único para cada libro. Se define como clave primaria y se genera automáticamente con `AUTO_INCREMENT`.

**título, autor:** guardan el nombre del libro y su autor, con un límite de 100 caracteres.

**genero:** indica el tipo de libro (ejemplo: ficción, ciencia, historia), con un máximo de 50 caracteres.

**publicacion:** almacena la fecha de publicación del libro en formato `DATE`.

**cantidad\_disponible:** registra cuántas copias hay disponibles para préstamo.

**calificacion:** permite guardar una puntuación (por ejemplo, de 1 a 5 estrellas) asignada al libro.

## 2. Tabla usuarios

La tabla usuarios guarda los datos de las personas registradas en la biblioteca.

**id\_usuario:** es el identificador único del usuario, definido como clave primaria y autoincremental.

**nombre:** almacena el nombre completo del usuario.

**direccion:** guarda la dirección física de la persona (hasta 150 caracteres).

**telefono:** permite registrar un número de contacto.

### 3. Tabla prestamos

La tabla prestamos gestiona los movimientos de préstamo de libros entre la biblioteca y los usuarios.

**id\_prestamo:** identificador único para cada préstamo.

**id\_libro:** referencia al libro prestado. Es una clave foránea que se vincula con libros(id\_libro).

**id\_usuario:** referencia al usuario que realiza el préstamo. También es una clave foránea hacia usuarios(id\_usuario).

**fecha\_prestamo:** indica la fecha en que se entregó el libro.

**fecha\_devolucion:** señala la fecha pactada para devolverlo.

**devuelto:** es un valor booleano (TRUE o FALSE) que muestra si el libro fue devuelto o no.

Esta tabla conecta directamente a libros y usuarios, y permite controlar quién tiene qué libro y en qué fechas.

### 4. Tabla reportes

La tabla reportes está pensada para guardar resultados de análisis o informes generados dentro del sistema.

**id\_reporte:** identificador único para cada reporte.

**tipo\_reporte:** especifica qué tipo de informe es (ejemplo: "Libros más prestados", "Usuarios morosos").

**fecha\_generacion:** registra la fecha en que se generó el reporte.

**descripcion:** guarda un texto explicativo sobre el reporte.

**resultado:** almacena el contenido del reporte, por ejemplo un resumen en texto o datos estadísticos.

**usuario\_genero:** indica quién fue el usuario que generó el reporte.