



GESTIÓN DE PRÉSTAMOS EN BIBLIOTECA

Fundamentos de base de datos



11 DE ABRIL DE 2025

DAMIAN ALEXANDER LUGO AGUILAR- 23490386
Prof. Jose Ramon Bogarin Valenzuela

Objetivo

La biblioteca desea optimizar el seguimiento de sus libros y usuarios. Para ello, necesita resolver los siguientes desafíos utilizando SQL. Nuestros querys iniciales quedarían así:

```
3 CREATE TABLE libros (
4     id SERIAL PRIMARY KEY,
5     titulo VARCHAR(150) NOT NULL,
6     autor VARCHAR(100),
7     anio_publicacion INT,
8     genero VARCHAR(50)
9 );
10 -- Tabla de usuarios
11 CREATE TABLE usuarios (
12     id SERIAL PRIMARY KEY,
13     nombre VARCHAR(100),
14     email VARCHAR(100) UNIQUE,
15     fecha_registro DATE
16 );
17 -- Tabla de préstamos
18 CREATE TABLE prestamos (
19     id SERIAL PRIMARY KEY,
20     id_libro INT REFERENCES libros(id),
21     id_usuario INT REFERENCES usuarios(id),
22     fecha_prestamo DATE,
23     fecha_devolucion DATE
24 );
```

```
1 -- Insertar datos en libros
2 INSERT INTO libros (titulo, autor, anio_publicacion, genero) VALUES
3 ('Cien años de soledad', 'Gabriel García Márquez', 1967, 'Realismo Mágico'),
4 ('1984', 'George Orwell', 1949, 'Distopía'),
5 ('El Principito', 'Antoine de Saint-Exupéry', 1943, 'Fábula'),
6 ('Fahrenheit 451', 'Ray Bradbury', 1953, 'Ciencia Ficción');
7 -- Insertar datos en usuarios
8 INSERT INTO usuarios (nombre, email, fecha_registro) VALUES
9 ('Laura Méndez', 'laura@example.com', '2024-10-01'),
10 ('Carlos Pérez', 'carlos@example.com', '2024-11-15'),
11 ('Andrea Gómez', 'andrea@example.com', '2025-01-20');
12 -- Insertar datos en préstamos
13 INSERT INTO prestamos (id_libro, id_usuario, fecha_prestamo, fecha_devolucion) VALUES
14 (1, 1, '2025-03-01', '2025-03-15'),
15 (2, 1, '2025-03-20', NULL),
16 (3, 2, '2025-03-05', '2025-03-25'),
17 (4, 3, '2025-04-01', NULL);
```

Parte 1: Estructura (LDD)

- 1. Agrega una columna telefono a la tabla usuarios.

```
2 --Agrega una columna telefono a la tabla usuarios.
3 ALTER TABLE usuarios
4 ADD telefono INT;
```


También agregamos los números para cada usuario

```
7 UPDATE usuarios
8 SET telefono= 123458
9 WHERE nombre='Andrea Gómez';
```

	id [PK] integer	nombre character varying (100)	email character varying (100)	fecha_registro date	telefono integer
1	1	Laura Méndez	laura@example.com	2024-10-01	123456
2	2	Carlos Pérez	carlos@example.com	2024-11-15	123457
3	3	Andrea Gómez	andrea@example.com	2025-01-20	123458

2. Cambia el tipo de dato de ani_publicacion en la tabla libros a SMALLINT.

```
10 --Cambia el tipo de dato de anio_publicacion en la tabla libros a SMALLINT.
11 ALTER TABLE libros
12 ALTER COLUMN anio_publicacion TYPE SMALLINT;
```



public
libros
id serial
titulo character varying(150)
autor character varying(100)
anio_publicacion smallint
genero character varying(50)

Parte 2: Manipulación (LMD)

1. Registra que “Andrea Gómez” ha devuelto el libro “Fahrenheit 451” hoy.

```
16 --Registra que “Andrea Gómez” ha devuelto el libro “Fahrenheit 451” hoy.
17 UPDATE prestamos
18 SET fecha_devolucion='2025-04-11'
19 WHERE id_usuario=3; --ID de Andrea Gómez
```

2. Cambia el género del libro “1984” a "Ciencia Ficción".

```
21 --Cambia el género del libro “1984” a "Ciencia Ficción".
22 UPDATE libros
23 SET genero= 'Ciencia Ficción'
24 WHERE titulo= '1984';
```

4	2	1984	George Orwell	1949	Ciencia Ficción
---	---	------	---------------	------	-----------------

3. Elimina el registro de préstamo del libro “El Principito” realizado por “Carlos Pérez”.

```
27 DELETE FROM prestamos
28 WHERE id_usuario=2;
```

Parte 3: Consultas (CLE)

1. Lista los libros que aún no han sido devueltos.

```
32 --Lista los libros que aún no han sido devueltos.
33 SELECT l.titulo AS libro, p.fecha_devolucion
34 FROM libros l INNER JOIN prestamos p
35 ON l.id=p.id_libro
36 WHERE p.fecha_devolucion IS NULL;
```

	libro character varying (150)	fecha_devolucion date
1	1984	[null]

2. Muestra cuántos libros ha prestado cada usuario.

```
37 --Muestra cuántos libros ha prestado cada usuario.
38 v SELECT u.nombre, COUNT (p.id_usuario) AS libros_prestados
39 FROM usuarios u INNER JOIN prestamos p
40 ON u.id=p.id_usuario
41 GROUP BY u.nombre, p.id_usuario
42 ORDER BY p.id_usuario ASC;
```

	nombre character varying (100) 🔒	libros_prestados bigint 🔒
1	Laura Méndez	2
2	Andrea Gómez	1

3. Consulta los libros más antiguos (anteriores al año 1950).

```
43 --Consulta los libros más antiguos (anteriores al año 1950).
44 v SELECT titulo, anio_publicacion
45 FROM libros
46 WHERE anio_publicacion<1950;
```

	titulo character varying (150) 🔒	anio_publicacion smallint 🔒
1	El Principito	1943
2	1984	1949

4. Calcula el promedio de duración (en días) de los préstamos devueltos.

```
47 --Calcula el promedio de duración (en días) de los préstamos devueltos.
48 v SELECT AVG(p.fecha_devolucion-p.fecha_prestamo) AS duracion_prestamo, u.nombre, l.titulo AS libro
49 FROM usuarios u INNER JOIN prestamos p
50 ON u.id=p.id_usuario
51 INNER JOIN libros l
52 ON l.id=p.id_libro
53 WHERE p.fecha_devolucion IS NOT NULL
54 GROUP BY u.nombre, l.titulo
55 ORDER BY duracion_prestamo DESC;
```

	duracion_prestamo numeric 🔒	nombre character varying (100) 🔒	libro character varying (150) 🔒
1	14.0000000000000000	Laura Méndez	Cien años de soledad
2	10.0000000000000000	Andrea Gómez	Fahrenheit 451

5. Lista todos los libros con su estado actual: “Prestado” o “Disponible”.

```
56 --Lista todos los libros con su estado actual: “Prestado” o “Disponible”.
57 v SELECT l.titulo AS libro, CASE
58     WHEN p.fecha_devolucion IS NULL THEN 'DISPONIBLE' ELSE 'PRESTADO'
59 END AS disponibilad
60 FROM libros l INNER JOIN prestamos p
61 ON l.id=p.id_libro
62 ORDER BY l.titulo;
```

	libro character varying (150) 🔒	disponibilad text 🔒
1	1984	DISPONIBLE
2	Cien años de soledad	PRESTADO
3	Fahrenheit 451	PRESTADO