

Instituto Tecnológico de Mexicali



Ingeniería Sistemas Computacionales

Fundamentos de Base de Datos

Tema:

“Gestión de Préstamos en Biblioteca”

Estudiante:

Andrés Mojica Medina

No. De control: C21490782

Docente:

José Ramón Bogarín Valenzuela

Mexicali, B.C., 11 de abril de 2025.

Query original

```
-- Crear tablas para sistema de biblioteca
```

```
-- Tabla de libros
```

```
CREATE TABLE libros (  
    id SERIAL PRIMARY KEY,  
    titulo VARCHAR(150) NOT NULL,  
    autor VARCHAR(100),  
    anio_publicacion INT,  
    genero VARCHAR(50)  
);
```

```
-- Tabla de usuarios
```

```
CREATE TABLE usuarios (  
    id SERIAL PRIMARY KEY,  
    nombre VARCHAR(100),  
    email VARCHAR(100) UNIQUE,  
    fecha_registro DATE  
);
```

```
-- Tabla de préstamos
```

```
CREATE TABLE prestamos (  
    id SERIAL PRIMARY KEY,  
    id_libro INT REFERENCES libros(id),  
    id_usuario INT REFERENCES usuarios(id),  
    fecha_prestamo DATE,
```

```
    fecha_devolucion DATE
);

-- Insertar datos en libros

INSERT INTO libros (titulo, autor, anio_publicacion, genero) VALUES
('Cien años de soledad', 'Gabriel García Márquez', 1967, 'Realismo Mágico'),
('1984', 'George Orwell', 1949, 'Distopía'),
('El Principito', 'Antoine de Saint-Exupéry', 1943, 'Fábula'),
('Fahrenheit 451', 'Ray Bradbury', 1953, 'Ciencia Ficción');

-- Insertar datos en usuarios

INSERT INTO usuarios (nombre, email, fecha_registro) VALUES
('Laura Méndez', 'laura@example.com', '2024-10-01'),
('Carlos Pérez', 'carlos@example.com', '2024-11-15'),
('Andrea Gómez', 'andrea@example.com', '2025-01-20');

-- Insertar datos en préstamos

INSERT INTO prestamos (id_libro, id_usuario, fecha_prestamo,
fecha_devolucion) VALUES
(1, 1, '2025-03-01', '2025-03-15'),
(2, 1, '2025-03-20', NULL),
(3, 2, '2025-03-05', '2025-03-25'),
(4, 3, '2025-04-01', NULL);
```

Problema a resolver: "Gestión de Préstamos en Biblioteca"

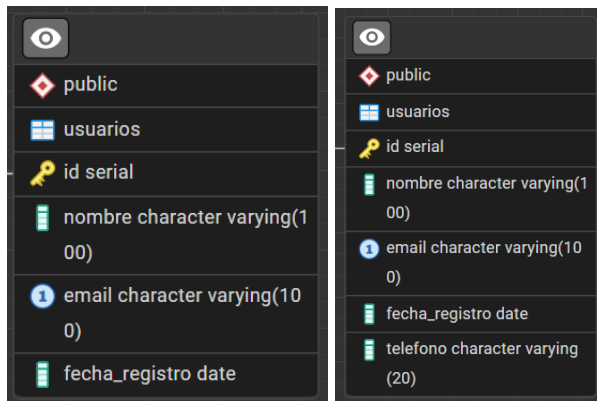
Objetivo:

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

Parte 1: Estructura (LDD)

1. Agrega una columna telefono a la tabla usuarios.

```
-- 1. Agregar columna telefono a la tabla usuarios
-- ALTER TABLE ADD COLUMN: Se utiliza para agregar una nueva columna a una tabla existente
ALTER TABLE usuarios
ADD COLUMN telefono VARCHAR(20);
```



Antes y después del cambio

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

```
-- 2. Cambiar tipo de dato de anio_publicacion a SMALLINT
-- ALTER TABLE ALTER COLUMN TYPE: Permite cambiar el tipo de datos de una columna existente
ALTER TABLE libros
ALTER COLUMN anio_publicacion TYPE SMALLINT;
```

Parte 2: Manipulación (LMD)

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

```
-- 1. Registrar que “Andrea Gómez” ha devuelto el libro “Fahrenheit 451” hoy
-- UPDATE JOIN WHERE: Se actualiza un valor usando condiciones que vinculan varias tablas
UPDATE prestamos
SET fecha_devolucion = CURRENT_DATE
WHERE id_libro = (SELECT id FROM libros WHERE titulo = 'Fahrenheit 451')
AND id_usuario = (SELECT id FROM usuarios WHERE nombre = 'Andrea Gómez');
```

	usuario character varying (100)	libro character varying (150)	fecha_prestamo date	fecha_devolucion date
1	Laura Méndez	Cien años de soledad	2025-03-01	2025-03-15
2	Carlos Pérez	El Principito	2025-03-05	2025-03-25
3	Laura Méndez	1984	2025-03-20	[null]
4	Andrea Gómez	Fahrenheit 451	2025-04-01	[null]













Antes

Data Output Messages Notifications				
	usuario character varying (100)	libro character varying (150)	fecha_prestamo date	fecha_devolucion date
1	Laura Méndez	Cien años de soledad	2025-03-01	2025-03-15
2	Carlos Pérez	El Principito	2025-03-05	2025-03-25
3	Laura Méndez	1984	2025-03-20	[null]
4	Andrea Gómez	Fahrenheit 451	2025-04-01	2025-04-11




Después

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

```
-- 2. Cambiar el género del libro “1984” a "Ciencia Ficción"  
-- UPDATE SET WHERE: Modifica una fila específica según una condición  
UPDATE libros  
SET genero = 'Ciencia Ficción'  
WHERE titulo = '1984';
```

Data Output Messages Notifications			
         SQL			
	id [PK] integer 	titulo character varying (150) 	genero character varying (50) 
1	1	Cien años de soledad	Realismo Mágico
2	2	1984	Distopía
3	3	El Principito	Fábula
4	4	Fahrenheit 451	Ciencia Ficción

Antes

Data Output Messages Notifications			
         SQL			
	id [PK] integer 	titulo character varying (150) 	genero character varying (50) 
1	1	Cien años de soledad	Realismo Mágico
2	3	El Principito	Fábula
3	4	Fahrenheit 451	Ciencia Ficción
4	2	1984	Ciencia Ficción

Despues

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

```
-- 3. Eliminar el registro de préstamo del libro “El Principito” realizado por “Carlos Pérez”
-- DELETE FROM WHERE: Elimina registros que cumplan ciertas condiciones
DELETE FROM prestamos
WHERE id_libro = (SELECT id FROM libros WHERE titulo = 'El Principito')
AND id_usuario = (SELECT id FROM usuarios WHERE nombre = 'Carlos Pérez');
```

Data Output Messages Notifications						
SQL						
	id	usuario	libro	fecha_prestamo	fecha_devolucion	
	integer	character varying (100)	character varying (150)	date	date	
1	1	Laura Méndez	Cien años de soledad	2025-03-01	2025-03-15	
2	2	Laura Méndez	1984	2025-03-20	[null]	
3	3	Carlos Pérez	El Principito	2025-03-05	2025-03-25	
4	4	Andrea Gómez	Fahrenheit 451	2025-04-01	2025-04-11	

Antes

Data Output Messages Notifications						
SQL						
	id	usuario	libro	fecha_prestamo	fecha_devolucion	
	integer	character varying (100)	character varying (150)	date	date	
1	1	Laura Méndez	Cien años de soledad	2025-03-01	2025-03-15	
2	2	Laura Méndez	1984	2025-03-20	[null]	
3	4	Andrea Gómez	Fahrenheit 451	2025-04-01	2025-04-11	

Despues

Parte 3: Consultas (CLE)

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

```
111 -- 1. Lista los libros que aún no han sido devueltos
112 -- SELECT WHERE IS NULL: Consulta registros con valor nulo en un campo
113 ✓ SELECT l.titulo, u.nombre AS usuario, p.fecha_prestamo
114 FROM prestamos p
115 JOIN libros l ON p.id_libro = l.id
116 JOIN usuarios u ON p.id_usuario = u.id
117 WHERE p.fecha_devolucion IS NULL;
118
```

Data Output Messages Notifications

	titulo character varying (150) 🔒	usuario character varying (100) 🔒	fecha_prestamo date 🔒
1	1984	Laura Méndez	2025-03-20

2. Muestra cuántos libros ha prestado cada usuario.

```
119 -- 2. Muestra cuántos libros ha prestado cada usuario
120 -- GROUP BY COUNT: Agrupa resultados y cuenta ocurrencias
121 ✓ SELECT u.nombre, COUNT(p.id) AS cantidad_prestamos
122 FROM usuarios u
123 LEFT JOIN prestamos p ON u.id = p.id_usuario
124 GROUP BY u.nombre;
125
```

Data Output Messages Notifications

	nombre character varying (100) 🔒	cantidad_prestamos bigint 🔒
1	Andrea Gómez	1
2	Carlos Pérez	0
3	Laura Méndez	2

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

```
126 -- 3. Consulta los libros más antiguos (anteriores al año 1950)
127 -- SELECT WHERE: Filtra registros por condición específica
128 SELECT titulo, anio_publicacion
129 FROM libros
130 WHERE anio_publicacion < 1950;
131
```

Data Output Messages Notifications

	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.

```
132 -- 4. Calcula el promedio de duración (en días) de los préstamos devueltos
133 -- AVG directo sobre la resta de fechas tipo DATE
134 SELECT ROUND(AVG(fecha_devolucion - fecha_prestamo)) AS promedio_dias
135 FROM prestamos
136 WHERE fecha_devolucion IS NOT NULL;
137
```

Data Output Messages Notifications

	promedio_dias numeric
1	12

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

```

138 -- 5. Lista todos los libros con su estado actual: "Prestado" o "Disponible"
139 -- CASE WHEN + EXISTS: Evalúa si un libro tiene un préstamo sin devolver
140 SELECT l.titulo,
141        CASE
142            WHEN EXISTS (
143                SELECT 1 FROM prestamos p
144                WHERE p.id_libro = l.id AND p.fecha_devolucion IS NULL
145            ) THEN 'Prestado'
146            ELSE 'Disponible'
147        END AS estado
148 FROM libros l;
149

```

Data Output Messages Notifications



	titulo character varying (150) 🔒	estado text 🔒
1	Cien años de soledad	Disponible
2	El Principito	Disponible
3	Fahrenheit 451	Disponible
4	1984	Prestado