

INSTITUTO TECNOLOGICO DE MEXICALI

Carrera:
ING. en Sistemas.

Materia:
Fundamento de base de datos.

Alumno:
Marin Salazar Juan Sebastian 22490423.

Correo Institucional:
a22490423@itmexicali.edu.mx

Profesor:
Jose Ramon Bogarin Valenzuela

Fecha de entrega:
11 de Abril del 2025

Mexicali, Baja California a 11 de Abril del 2025.

Querys del profe

-- 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)

- Agrega una columna telefono a la tabla usuarios.

R=

```
ALTER TABLE usuarios
```

```
ADD telefono VARCHAR(20);
```

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

R=

```
ALTER TABLE libros
```

```
ALTER COLUMN anio_publicacion TYPE SMALLINT;
```

Parte 2: Manipulación (LMD)

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

R=

```
UPDATE prestamos P
```

```
SET fecha_devolucion = CURRENT_DATE
```

```
FROM usuarios U
```

```
INNER JOIN prestamos Pr ON Pr.id_usuario = U.id
```

```
INNER JOIN libros L ON Pr.id_libro = L.id
```

```
WHERE P.id = Pr.id
```

```
AND U.nombre = 'Andrea Gómez'
```

```
AND L.titulo = 'Fahrenheit 451'
```

```
AND P.fecha_devolucion IS NULL;
```

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

R=

```
UPDATE libros
SET genero = 'Ciencia Ficción'
WHERE titulo = '1984';
```

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

R=

```
DELETE FROM prestamos P
USING usuarios U
INNER JOIN prestamos Pr ON Pr.id_usuario = U.id
INNER JOIN libros L ON Pr.id_libro = L.id
WHERE P.id = Pr.id
AND U.nombre = 'Carlos Pérez'
AND L.titulo = 'El Principito';
```

Parte 3: Consultas (CLE)

- Lista los libros que aún no han sido devueltos.

R=

```
SELECT L.titulo, L.autor, U.nombre AS nombre_usuario, P.fecha_prestamo
FROM usuarios U
INNER JOIN prestamos P ON P.id_usuario = U.id
INNER JOIN libros L ON P.id_libro = L.id
WHERE P.fecha_devolucion IS NULL;
```

- Muestra cuántos libros ha prestado cada usuario.

R=

```
SELECT U.nombre, COUNT(P.id) AS total_prestamos
FROM usuarios U
INNER JOIN prestamos P ON U.id = P.id_usuario
GROUP BY U.nombre
ORDER BY total_prestamos DESC;
```

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

R=

```
SELECT L.titulo, L.autor, L.anio_publicacion, L.genero
FROM libros L
WHERE L.anio_publicacion < 1950
ORDER BY L.anio_publicacion ASC;
```

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

R=

```
SELECT AVG(P.fecha_devolucion - P.fecha_prestamo) AS promedio_dias_prestamo
FROM prestamos P
WHERE P.fecha_devolucion IS NOT NULL;
```

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

R=

```
SELECT L.titulo, L.autor,
CASE
WHEN EXISTS (
SELECT 1
FROM prestamos P
WHERE P.id_libro = L.id AND P.fecha_devolucion IS NULL)
THEN 'Prestado'
ELSE 'Disponible'
END AS estado
FROM libros L
ORDER BY estado DESC, L.titulo;
```

Resultados

Query

Query History

1

2

3

4

5

6

SELECT L.titulo, L.autor, U.nombre AS nombre_usuario, P.fecha_prestamo

FROM usuarios U

INNER JOIN prestamos P ON P.id_usuario = U.id

INNER JOIN libros L ON P.id_libro = L.id

WHERE P.fecha_devolucion IS NULL;

Data Output

Messages

Notifications

SQL

Showing

	titulo character varying (150)	autor character varying (100)	nombre_usuario character varying (100)	fecha_prestamo date
1	1984	George Orwell	Laura Méndez	2025-03-20

Query

Query History

1

2

3

4

5

6

SELECT U.nombre, COUNT(P.id) AS total_prestamos

FROM usuarios U

INNER JOIN prestamos P ON U.id = P.id_usuario

GROUP BY U.nombre

ORDER BY total_prestamos DESC;

Data Output

Messages

Notifications

SQL

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

Query

Query History

1

2

3

4

5

SELECT L.titulo, L.autor, L.anio_publicacion, L.genero

FROM libros L

WHERE L.anio_publicacion < 1950

ORDER BY L.anio_publicacion ASC;

Data Output

Messages

Notifications

SQL

Showing

	titulo character varying (150)	autor character varying (100)	anio_publicacion smallint	genero character varying (50)
1	El Principito	Antoine de Saint-Exupéry	1943	Fábula
2	1984	George Orwell	1949	Ciencia Ficción

Query

Query History

1

2

3

4

SELECT AVG(P.fecha_devolucion - P.fecha_prestamo) AS promedio_dias_prestamo

FROM prestamos P

WHERE P.fecha_devolucion IS NOT NULL;

Data Output

Messages

Notifications

Showing rows: 1

promedio_dias_prestamo

numeric

1

12.0000000000000000

Query

Query History

1

2

3

4

5

6

7

8

9

10

11

SELECT L.titulo, L.autor,

CASE

WHEN EXISTS (

SELECT 1

FROM prestamos P

WHERE P.id_libro = L.id AND P.fecha_devolucion IS NULL)

THEN 'Prestado'

ELSE 'Disponible'

END AS estado

FROM libros L

ORDER BY estado DESC, L.titulo;

Data Output

Messages

Notifications

Showing rows: 4

titulo

character varying (150)

autor

character varying (100)

estado

text

1

1984

George Orwell

Prestado

2

Cien años de soledad

Gabriel García Márquez

Disponible

3

El Principito

Antoine de Saint-Exupéry

Disponible

4

Fahrenheit 451

Ray Bradbury

Disponible