

1.1. Creación de tabla libros

```
1  -- Crear tabla libros
2  CREATE TABLE libros (
3      id SERIAL PRIMARY KEY,
4      titulo VARCHAR(100),
5      autor VARCHAR(100),
6      stock INTEGER
7  );
8
```

1.2. Creación de tabla usuarios

```
9  -- Crear tabla usuarios
10 CREATE TABLE usuarios (
11     id SERIAL PRIMARY KEY,
12     nombre VARCHAR(100),
13     email VARCHAR(100)
14 );
15
16 -- Crear tabla prestamos con claves foráneas
17 CREATE TABLE prestamos (
18     id SERIAL PRIMARY KEY,
19     id_libro INT REFERENCES libros(id),
20     id_usuario INT REFERENCES usuarios(id),
21     fecha DATE
22 );
```

Data Output Messages Notifications

CREATE TABLE

Total rows: Query complete 00:00:00.038

Activar Windows

Query returned successfully in 38 msec. X

CRLF Ln 9, Col 1

1.3. Creación de tabla prestamos

```
16 -- Crear tabla prestamos con claves foráneas
17 CREATE TABLE prestamos (
18     id SERIAL PRIMARY KEY,
19     id_libro INT REFERENCES libros(id),
20     id_usuario INT REFERENCES usuarios(id),
21     fecha DATE
22 );
```

Data Output Messages Notifications

CREATE TABLE

Total rows: Query complete 00:00:00.046

Activar Windows

Query returned successfully in 46 msec. X

CRLF Ln 16, Col 1

2.1. Inserción de los datos libros

```
26 -- Insertar libros
27 INSERT INTO libros (titulo, autor, stock) VALUES
28 ('Cien años de soledad', 'Gabriel García Márquez', 5),
29 ('1984', 'George Orwell', 4),
30 ('Don Quijote', 'Miguel de Cervantes', 3),
31 ('Rayuela', 'Julio Cortázar', 2),
32 ('El Principito', 'Antoine de Saint-Exupéry', 6);
33
34 -- Insertar usuarios
35 INSERT INTO usuarios (nombre, email) VALUES
36 ('Juan Pérez', 'juan@example.com'),
37 ('María López', 'maria@example.com'),
38 ('Carlos Ruiz', 'carlos@example.com');
39
40 -- Insertar préstamos
41 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
42 (1, 1, '2025-07-15'),
43 (2, 2, '2025-07-16'),
44 (3, 3, '2025-07-17');
45
```

Data Output Messages Notifications

INSERT 0 5

Activar Windows

Query returned successfully in 43 msec. X

Total rows: Query complete 00:00:00.040

CRLF Ln 45, Col 1

2.2. inserción de los datos usuarios

```
34 -- Insertar usuarios
35 INSERT INTO usuarios (nombre, email) VALUES
36 ('Juan Pérez', 'juan@example.com'),
37 ('María López', 'maria@example.com'),
38 ('Carlos Ruiz', 'carlos@example.com');
39
40 -- Insertar préstamos
41 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
42 (1, 1, '2025-07-15'),
43 (2, 2, '2025-07-16'),
44 (3, 3, '2025-07-17');
45
```

Data Output Messages Notifications

INSERT 0 3

Total rows: Query complete 00:00:00.042

Activar Windows

Query returned successfully in 42 msec. X

CRLF Ln 34, Col 1

2.3. inserción de los datos prestamos

```
40 -- Insertar préstamos
41 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
42 (1, 1, '2025-07-15'),
43 (2, 2, '2025-07-16'),
44 (3, 3, '2025-07-17');
45
```

Data Output Messages Notifications

INSERT 0 3

Total rows: Query complete 00:00:00.050

Activar Windows

Query returned successfully in 50 msec. X

CRLF Ln 40, Col 1

3.1. creación de roles

```
46 --3. creacion de roles y permisos de lectura en tabla libros
47 -- Crear rol lector_bc
48 CREATE ROLE lector_bc LOGIN PASSWORD 'lectorpwd';
49
50 -- Dar permisos de solo lectura sobre libros
51 GRANT CONNECT ON DATABASE biblioteca_bc TO lector_bc;
52 GRANT USAGE ON SCHEMA public TO lector_bc;
53 GRANT SELECT ON libros TO lector_bc;
54
```

Data Output Messages Notifications

CREATE ROLE

Total rows: Query complete 00:00:00.052

Activar Windows

Query returned successfully in 52 msec. X

CRLF Ln 47, Col 1

3.2. asignación de los permisos al rol

```
50 -- Dar permisos de solo lectura sobre libros
51 GRANT CONNECT ON DATABASE biblioteca_bc TO lector_bc;
52 GRANT USAGE ON SCHEMA public TO lector_bc;
53 GRANT SELECT ON libros TO lector_bc;
54
```

Data Output Messages Notifications

GRANT

Total rows: Query complete 00:00:00.043

Activar Windows

Query returned successfully in 43 msec. X

CRLF Ln 50, Col 1

4.1. inicio de transacción con begin

```
55 --4. inicio de transaccion con begin
56 -- Iniciar transacción
57 BEGIN;
58
59 -- Insertar un préstamo válido
60 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
61 (1, 2, '2025-07-18');
62
63 -- Insertar un préstamo con error (libro id 999 no existe)
64 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
65 (999, 2, '2025-07-18');
66
67 -- Como hay error, hacemos rollback
68 ROLLBACK;
69
70 -- Volvemos a iniciar y corregimos
71 BEGIN;
72
73 -- Insertamos uno correcto
74 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
```

Data Output Messages Notifications

BEGIN

Activar Windows

Query returned successfully in 37 msec. X

Total rows: Query complete 00:00:00.037 CRLF Ln 56, Col 1

4.2. inserción de un préstamo valido

```
59 -- Insertar un préstamo válido
60 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
61 (1, 2, '2025-07-18');
62
63 -- Insertar un préstamo con error (libro id 999 no existe)
64 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
65 (999, 2, '2025-07-18');
66
67 -- Como hay error, hacemos rollback
68 ROLLBACK;
69
70 -- Volvemos a iniciar y corregimos
71 BEGIN;
72
73 -- Insertamos uno correcto
74 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
```

Data Output Messages Notifications

INSERT 0 1

Activar Windows

Query returned successfully in 44 msec. X

Total rows: Query complete 00:00:00.044 CRLF Ln 59, Col 1

4.3. inserción de préstamo con error intencional

```
63 -- Insertar un préstamo con error (libro id 999 no existe)
64 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
65 (999, 2, '2025-07-18');
66
```

Messages

ERROR: inserción o actualización en la tabla «prestamos»
viola la llave foránea «prestamos_id_libro_fkey»
La llave (id_libro)=(999) no está presente en la tabla
«libros».

SQL state: 23503

Detail: La llave (id_libro)=(999) no está presente en la tabla
«libros».

4.4. uso del ROLLBACK

```
67 -- Como hay error, hacemos rollback
68 ROLLBACK;
69
70 -- Volvemos a iniciar y corregimos
71 BEGIN;
72
73 -- Insertamos uno correcto
74 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
75 (2, 2, '2025-07-18');
76
77 -- Finalizamos correctamente
78 COMMIT;
79
```

Data Output Messages Notifications

ROLLBACK

Query returned successfully in 45 msec.

Activar Windows

✓ Query returned successfully in 45 msec. ✕

Total rows: Query complete 00:00:00.045 CRLF Ln 67, Col 1

4.5. realizamos begin e insertamos otro préstamo corregido

```
70 -- Volvemos a iniciar y corregimos
71 BEGIN;
72
73 -- Insertamos uno correcto
74 INSERT INTO prestamos (id_libro, id_usuario, fecha) VALUES
75 (2, 2, '2025-07-18');
76
77 -- Finalizamos correctamente
78 COMMIT;
79
```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 44 msec.

Activar Windows

✓ Query returned successfully in 44 msec. ✕

Total rows: Query complete 00:00:00.044 CRLF Ln 73, Col 1

4.6. finalización con COMMIT

```
77 -- Finalizamos correctamente
78 COMMIT;
79
```

Data Output Messages Notifications

COMMIT

Query returned successfully in 38 msec.

Activar Windows

✓ Query returned successfully in 38 msec. ✕

Total rows: Query complete 00:00:00.038 CRLF Ln 77, Col 1

CONCLUSION:

80 **SELECT * FROM LIBROS;**

Data Output Messages Notifications

Showing rows: 1 to 5 Page No: 1 of 1

	id [PK] integer	titulo character varying (100)	autor character varying (100)	stock integer
1	1	Cien años de soledad	Gabriel García Márquez	5
2	2	1984	George Orwell	4
3	3	Don Quijote	Miguel de Cervantes	3
4	4	Rayuela	Julio Cortázar	2
5	5	El Principito	Antoine de Saint-Exupéry	6

Total rows: 5 Query complete 00:00:00.271

Activar Windows
Verificar éxito: Successfully run! Total query runtime: 271 ms

1. ¿Qué aprendiste al aplicar todos los sublenguajes SQL en un solo ejercicio?

Aprendí cómo se integran los sublenguajes DDL (definición de tablas), DML (inserciones), DCL (permisos) y TCL (transacciones) en una solución real, mostrando cómo funcionan juntos para garantizar consistencia y seguridad en una base de datos.

2. ¿Qué errores lograste evitar usando ROLLBACK?

Evité que se guardara un préstamo con un id_libro inválido (999), lo que habría generado inconsistencia. Con ROLLBACK, todo lo que se intentó guardar después del error fue descartado.

3. ¿Para qué sirve crear usuarios con permisos limitados?

Sirve para proteger la base de datos de accesos no autorizados o acciones no deseadas, como borrar o modificar información crítica. El rol lector solo puede consultar datos, ideal para usuarios que solo necesitan visualizar.