



Student Name: ADITYA RAJ BHAKTA

Branch: AIT-CSE (AIML)

Semester: 4

Subject Name: Database Management System

UID: 24BAI70782

Section/Group: 24AIT_KRG2

Subject Code: 24CSH-298

Experiment

Experiment 1.1: Design and implementation of a Library Management System using PostgreSQL with DDL, DML and DCL commands.

Aim

The aim of this experiment is to design and implement a Library Management System database using PostgreSQL. The database is created using proper tables, primary keys, foreign keys and constraints. DML operations are performed and database security is implemented using roles and privileges.

Objective

The objective of this experiment is to gain practical knowledge of DDL, DML and DCL commands in PostgreSQL. It also helps in understanding how to create roles, grant permissions and revoke permissions to secure the database using role based access control.

Practical / Experiment Steps

1. Design the database structure for the Library Management System.
2. Create tables for books, members and issue records using DDL commands.
3. Apply primary keys, foreign keys and constraints to maintain data integrity.
4. Insert sample records into the tables using DML commands.
5. Update and delete records as required.
6. Create a database role named Librarian.
7. Grant required permissions like SELECT, INSERT and DELETE to the Librarian role.
8. Revoke permissions when needed to ensure database security.

Procedure of the Experiment

1. Start the system and log in to the computer.
2. Open pgAdmin and connect to PostgreSQL server.
3. Create a new database for the Library Management System.
4. Create tables such as Books, Members and Issue_Records using CREATE TABLE command.
5. Define primary keys and foreign keys while creating the tables.
6. Insert records into tables using INSERT command.
7. Update existing data using UPDATE command.
8. Delete unwanted records using DELETE command.
9. Create a role named Librarian with password using CREATE ROLE command.
10. Grant SELECT, INSERT and DELETE permissions to the Librarian role.
11. Revoke permissions using REVOKE command when required.
12. Execute all queries and verify the output.

CODE :

1. ADMIN

```
select CURRENT_USER;
```

```
CREATE TABLE book(  
id INT PRIMARY KEY,  
name VARCHAR(50) NOT NULL,  
authorName VARCHAR(50));
```

```
INSERT INTO book  
(id,name,authorName)  
VALUES (101,'your name','aditya'),  
(102,'last summer','ayush');
```



ALTER TABLE book

ADD COUNT INT CHECK(COUNT>=1);

UPDATE book

SET COUNT=3

WHERE ID=101;

SELECT * FROM book;

```
CREATE TABLE LIBRARY_VISITOR_USER(  
  USER_ID INT PRIMARY KEY,  
  USER_NAME VARCHAR(20) NOT NULL,  
  AGE INT CHECK (AGE>=17) NOT NULL,  
  EMAIL VARCHAR(50) UNIQUE NOT NULL  
)
```

```
ALTER TABLE LIBRARY_VISITOR_USER  
DROP COLUMN EMAIL;
```

```
ALTER TABLE LIBRARY_VISITOR_USER  
ADD EMAIL VARCHAR(50) UNIQUE NOT NULL;
```

```
INSERT INTO LIBRARY_VISITOR_USER  
(USER_ID,USER_NAME,AGE,EMAIL) VALUES  
(101,'ADITYA',18,'adityarajbhakta@gmail.com');
```

```
SELECT * FROM LIBRARY_VISITOR_USER;
```



```
CREATE TABLE BOOK_ISSUE(  
BOOK_ISSUE_ID INT PRIMARY KEY,  
BOOK_ID INT REFERENCES book(id),  
USER_ID INT REFERENCES LIBRARY_VISITOR_USER(USER_ID),  
BOOK_ISSUE_DATE DATE NOT NULL  
);
```

```
INSERT INTO BOOK_ISSUE (BOOK_ISSUE_ID,BOOK_ID, USER_ID,BOOK_ISSUE_DATE) VALUES  
(34899345,102,101,'2026-05-01');
```

```
SELECT * FROM BOOK_ISSUE;
```

```
CREATE ROLE LIBRARIAN1 WITH LOGIN PASSWORD 'ADITYA-RAJ-BHAKTA';
```

```
CREATE ROLE LIBRARIAN_1  
WITH LOGIN PASSWORD '12233445'
```

```
GRANT SELECT,INSERT,UPDATE,DELETE ON book TO LIBRARIAN1;  
GRANT SELECT,INSERT,UPDATE,DELETE ON LIBRARY_VISITOR_USER TO LIBRARIAN1;  
GRANT SELECT,INSERT,UPDATE,DELETE ON BOOK_ISSUE TO LIBRARIAN1;
```

```
REVOKE SELECT ,INSERT ,DELETE,UPDATE ON LIBRARY_VISITOR_USER FROM LIBRARIAN1  
REVOKE SELECT ,INSERT ,DELETE,UPDATE ON BOOKS FROM LIBRARIAN1  
REVOKE SELECT ,INSERT ,DELETE,UPDATE ON BOOK_ISSUE FROM LIBRARIAN1
```

2. LIBRARIAN1

```
SELECT * FROM book;  
select * from book_issue;  
select * from LIBRARY_VISITOR_USER;
```

```
INSERT INTO book VALUES(110,'can we meet ','rishu',2);  
INSERT INTO book VALUES(150,'hobit','piyush',7);
```

```
SELECT * FROM book_issue;  
SELECT * FROM LIBRARY_VISITOR_USER;
```

Learning Outcomes:

1. Understood the basics of **relational database design** using tables, keys, and relationships.
2. Learned to apply **primary key and foreign key constraints** to maintain data integrity.
3. Gained hands-on experience with **INSERT, UPDATE, and DELETE** operations safely.
4. Understood how **roles and privileges** control access to database objects.
5. Learned to use **GRANT and REVOKE** for implementing **read-only users**.
6. Practiced **ALTER TABLE and DROP TABLE** for managing database changes.

SCREENSHOTS

```
select CURRENT_USER;
```

```
//BOOKS TABLE  
CREATE TABLE book(  
id INT PRIMARY KEY,  
name VARCHAR(50) NOT NULL,  
authorName VARCHAR(50));  
INSERT INTO book (id,name,authorName) VALUES (101,'your name','aditya'),(102,'last summer','ayush');  
ALTER TABLE book ADD COUNT INT CHECK(COUNT>=1);  
SELECT * FROM book;  
UPDATE book SET COUNT=3 WHERE id=101;|
```

```
//VISITOR
CREATE TABLE LIBRARY_VISITOR_USER(
USER_ID INT PRIMARY KEY,
USER_NAME VARCHAR(20) NOT NULL,
AGE INT CHECK (AGE>=17) NOT NULL,
EMAIL VARCHAR(50) UNIQUE NOT NULL
);
INSERT INTO LIBRARY_VISITOR_USER (USER_ID,USER_NAME,AGE,EMAIL) VALUES (101,'ADITYA',
18,'adityarajbhakta@gmail.com');
SELECT * FROM LIBRARY_VISITOR_USER;
```

```
//BOOK ISSUE
CREATE TABLE BOOK_ISSUE(
BOOK_ISSUE_ID INT PRIMARY KEY,
BOOK_ID INT REFERENCES book(id),
USER_ID INT REFERENCES LIBRARY_VISITOR_USER(USER_ID),
BOOK_ISSUE_DATE DATE NOT NULL
);
INSERT INTO BOOK_ISSUE (BOOK_ISSUE_ID,BOOK_ID, USER_ID,BOOK_ISSUE_DATE) VALUES
(34899345,102,101,'2026-05-01');
SELECT * FROM BOOK_ISSUE;
```

```
//ROLE
CREATE ROLE LIBRARIAN1 WITH LOGIN PASSWORD 'ADITYA-RAJ-BHAKTA';
GRANT SELECT,INSERT,UPDATE,DELETE ON book TO LIBRARIAN1;
GRANT SELECT,INSERT,UPDATE,DELETE ON LIBRARY_VISITOR_USER TO LIBRARIAN1;
GRANT SELECT,INSERT,UPDATE,DELETE ON BOOK_ISSUE TO LIBRARIAN1;
REVOKE DELETE ON book FROM LIBRARIAN1;

SELECT * FROM book;
select * from book_issue;
select * from LIBRARY_VISITOR_USER;

INSERT INTO book VALUES(110,'can we meet','rishu',2);
INSERT INTO book VALUES(150,'hobit','hobit',7);

SELECT * FROM book_issue;
SELECT * FROM LIBRARY_VISITOR_USER;
```

	id [PK] integer	name character varying (50)	authorname character varying (50)	count integer
1	102	last summer	ayush	[null]
2	101	your name	aditya	3

	book_issue_id [PK] integer	book_id integer	user_id integer	book_issue_date date
1	34899345	102	101	2026-05-01

	user_id [PK] integer	user_name character varying (20)	age integer	email character varying (50)
1	101	ADITYA	18	adityarajbhakta@gmail.c...