



# Apex Institute of Technology

## Computer Science & Engineering

### Worksheet 1

**Name:** Trimann Kaur

**UID:** 24BAI70511

**Branch:** B.E. CSE (AIML)

**Section:** 24AIT\_KRG-G1

**Semester:** 4

**Date of Performance:** 09.01.2026

**Subject Name:** Database  
Management System

**Subject Code:** 24CSH-298

**AIM:** To design and implement a Library Management System database using appropriate tables, primary keys, foreign keys, and constraints, and to perform DML operations along with DCL commands such as role creation, privilege granting, and revoking to ensure database security.

#### **OBJECTIVES:**

1. To implement Data Definition Language (DDL) commands for creating, altering, and deleting database tables with appropriate constraints.
2. To apply Data Manipulation Language (DML) commands to insert, update, retrieve, and manage records while maintaining data integrity.
3. To understand Data Control Language (DCL) by creating user roles and managing database security through granting and revoking privileges.

#### **SOFTWARE REQUIREMENTS:**

- Database Management System
  - PostgreSQL
- Database Administration Tool
  - pgAdmin

#### **PRACTICAL/EXPERIMENT STEPS:**

1. A table named BOOKS was created to store book details such as Book ID, Book Name, and Author Name. Book\_ID is set as the primary key.
2. The table BOOKS was modified using ALTER command to add a new column BOOK\_COUNT with a check constraint to ensure that the BOOK\_COUNT is greater than 0 and is not null.
3. Records were inserted into the BOOKS table using INSERT commands and verified using SELECT queries.



# Apex Institute of Technology

## Computer Science & Engineering

4. A new table LIBRARY\_VISITORS was created with attributes USER\_ID, NAME, AGE and EMAIL. USER\_ID is set as the primary key and data is inserted in the table using INSERT command.
5. A new table BOOK\_ISSUE was created to maintain book issue records, implementing FOREIGN KEY relationships between BOOKS and LIBRARY\_VISITORS tables. USER\_ID from LIBRARY\_VISITORS table and BOOK\_ID from BOOKS are set as the foreign key.
6. The ALTER table command is used to add ISSUE\_DATE for issued books. UPDATE command is used to update the previous records of the table.
7. A new user role named LIBRARIAN was created, and permissions were granted using GRANT.
8. Permissions were later revoked using the REVOKE command to understand access control.
9. TRUNCATE and DROP commands were executed to observe the difference between removing records and deleting tables.

### **PROCEDURE:**

1. Open PostgreSQL and create a new database.
2. Execute the following command to create a table BOOKS.

```
CREATE TABLE BOOKS (
BOOK_ID INT PRIMARY KEY,
BOOK_NAME VARCHAR(20) NOT NULL,
AUTHOR_NAME VARCHAR(20) NOT NULL
)
```

3. Alter the table to add another attribute BOOK\_COUNT with a constraint.

```
ALTER TABLE BOOKS
ADD BOOK_COUNT INT CHECK(BOOK_COUNT > 0) NOT NULL
```

4. Insert data into the table.

```
INSERT INTO BOOKS VALUES(101, 'Harry Potter', 'Rowling', 3)
INSERT INTO BOOKS VALUES(102, 'The Alchemist', 'Paulo', 5);
```

5. Create a new table LIBRARY\_VISITORS using CREATE command

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

6. Insert data into this table.

```
INSERT INTO LIBRARY_VISITORS(USER_ID, NAME, AGE, EMAIL)
VALUES(501, 'vansh', 19, 'vansh@gmail.com')

INSERT INTO LIBRARY_VISITORS(USER_ID, NAME, AGE, EMAIL)
VALUES(502, 'ansh', 19, 'ansh@gmail.com')
```



# Apex Institute of Technology

## Computer Science & Engineering

7. Create a new table BOOK\_ISSUE using the following command, and insert data into it.

```
CREATE TABLE BOOK_ISSUE(  
BOOK_ISSUE_ID INT PRIMARY KEY,  
BOOK_ID INT NOT NULL,  
USER_ID INT NOT NULL,  
FOREIGN KEY(USER_ID) REFERENCES LIBRARY_VISITORS(USER_ID),  
FOREIGN KEY(BOOK_ID) REFERENCES BOOKS(BOOK_ID)  
)
```

```
INSERT INTO BOOK_ISSUE VALUES(10001, 101, 501)
```

8. Use the ALTER and UPDATE command as follows to update the previous records.

```
ALTER TABLE BOOK_ISSUE  
ADD ISSUE_DATE DATE
```

```
UPDATE BOOK_ISSUE  
SET ISSUE_DATE='2026-01-08'  
WHERE BOOK_ID=101
```

9. Execute the following command to create a new role of LIBRARIAN.

```
CREATE ROLE LIBRARIAN  
WITH LOGIN PASSWORD '12345678'
```

10. Use GRANT command to grant access to the LIBRARIAN.

```
GRANT SELECT, INSERT, DELETE, UPDATE ON BOOKS TO LIBRARIAN  
GRANT SELECT, INSERT, DELETE, UPDATE ON BOOK_ISSUE TO LIBRARIAN  
GRANT SELECT, INSERT, DELETE, UPDATE ON LIBRARY_VISITORS TO LIBRARIAN
```

11. Demonstrate the use of REVOKE using the following command.

```
REVOKE SELECT, INSERT, DELETE, UPDATE ON BOOKS, BOOK_ISSUE, LIBRARY_VISITORS FROM LIBRARIAN
```

12. Use TRUNCATE to delete all records in the table, and DROP to delete the table.

```
TRUNCATE TABLE BOOK_ISSUE | DROP TABLE BOOK_ISSUE .....
```

### I/O ANALYSIS:

- Create table BOOKS.



# Apex Institute of Technology

## Computer Science & Engineering

```
CREATE TABLE BOOKS (
BOOK_ID INT PRIMARY KEY,
BOOK_NAME VARCHAR(20) NOT NULL,
AUTHOR_NAME VARCHAR(20) NOT NULL
)
```

| book_id<br>[PK] integer ↴ | book_name<br>character varying (20) ↴ | author_name<br>character varying (20) ↴ |
|---------------------------|---------------------------------------|---|
| 1                         | 101 Harry Potter                      | Rowling                                 |
| 2                         | 102 The Alchemist                     | Paulo                                   |

- Alter and insert into table BOOKS.

```
ALTER TABLE BOOKS
ADD BOOK_COUNT INT CHECK(BOOK_COUNT > 0) NOT NULL
```

```
INSERT INTO BOOKS VALUES(101, 'Harry Potter', 'Rowling', 3)
INSERT INTO BOOKS VALUES(102, 'The Alchemist', 'Paulo', 5);
```

| book_id<br>[PK] integer ↴ | book_name<br>character varying (20) ↴ | author_name<br>character varying (20) ↴ | book_count<br>integer ↴ |
|---------------------------|---------------------------------------|---|-------------------------|
| 1                         | 101 Harry Potter                      | Rowling                                 | 3                       |
| 2                         | 102 The Alchemist                     | Paulo                                   | 5                       |

- Create table LIBRARY\_VISITORS.

```
CREATE TABLE BOOK_ISSUE(
BOOK_ISSUE_ID INT PRIMARY KEY,
BOOK_ID INT NOT NULL,
USER_ID INT NOT NULL,
FOREIGN KEY(USER_ID) REFERENCES LIBRARY_VISITORS(USER_ID),
FOREIGN KEY(BOOK_ID) REFERENCES BOOKS(BOOK_ID)
)
```

| user_id<br>[PK] integer ↴ | name<br>character varying (20) ↴ | age<br>integer ↴ | email<br>character varying (20) ↴ |
|---------------------------|----------------------------------|------------------|-----------------------------------|
| 1                         | 501 vansk                        | 19               | vansk@gmail.com                   |
| 2                         | 502 ansh                         | 19               | ansh@gmail.com                    |

- Insert into LIBRARY\_VISITORS.

```
INSERT INTO LIBRARY_VISITORS(USER_ID, NAME, AGE, EMAIL)
VALUES(501, 'vansk', 19, 'vansk@gmail.com')
INSERT INTO LIBRARY_VISITORS(USER_ID, NAME, AGE, EMAIL)
VALUES(502, 'ansh', 19, 'ansh@gmail.com')
```

| user_id<br>[PK] integer ↴ | name<br>character varying (20) ↴ | age<br>integer ↴ | email<br>character varying (20) ↴ |
|---------------------------|----------------------------------|------------------|-----------------------------------|
| 1                         | 501 vansk                        | 19               | vansk@gmail.com                   |
| 2                         | 502 ansh                         | 19               | ansh@gmail.com                    |

- Create table BOOK\_ISSUE with foreign key. Alter and update it.

|   | book_issue_id<br>[PK] integer ↴ | book_id<br>integer ↴ | user_id<br>integer ↴ |
|---|---------------------------------|----------------------|----------------------|
| 1 | 10001                           | 101                  | 501                  |

- Alter table BOOK\_ISSUE.

|   | book_issue_id<br>[PK] integer ↴ | book_id<br>integer ↴ | user_id<br>integer ↴ | issue_date<br>date ↴ |
|---|---------------------------------|----------------------|----------------------|----------------------|
| 1 | 10001                           | 101                  | 501                  | [null]               |

- Update BOOK\_ISSUE.



# Apex Institute of Technology

## Computer Science & Engineering

|   | book_issue_id<br>[PK] integer ↴ | book_id<br>integer ↴ | user_id<br>integer ↴ | issue_date<br>date ↴ |
|---|---------------------------------|----------------------|----------------------|----------------------|
| 1 | 10001                           | 101                  | 501                  | 2026-01-08           |

- Creating a new role, and granting access.

```
GRANT

Query returned successfully in 116 msec.
```

- Revoke access.

```
REVOKE

Query returned successfully in 84 msec.          ERROR: permission denied for table books
                                                SQL state: 42501
```

- Truncate table BOOK\_ISSUE.

|  | book_issue_id<br>[PK] integer ↴ | book_id<br>integer ↴ | user_id<br>integer ↴ | issue_date<br>date ↴ |
|--|---------------------------------|----------------------|----------------------|----------------------|
|  |                                 |                      |                      |                      |

- Drop table BOOK\_ISSUE.

```
ERROR: relation "book_issue" does not exist
```

### LEARNING OUTCOMES:

- Understanding of creating and managing database structures using DBL commands.
- Inserting, updating, and retrieving data using DML queries while maintaining data integrity.
- Knowledge of controlling database access by creating roles and applying DCL commands.