Lab Session 03

3. Database Creation & Management

Aim/Objective:

The aim of this experiment is to familiarize students with the To download, install the PostgreSQL and practical implementation of creating database objects, defining their structure, and apply constraints using DDL operations.

Description:

PostgreSQL is a popular relational database management system (RDBMS). PostgreSQL is not controlled by any corporation or other private entity and the source code is available free of charge. PostgreSQL runs on all major operating systems, including Linux, UNIX and Windows. It supports text, images, sounds, and video, and includes programming interfaces for C / C++, Java, Perl, Python, Ruby, Tcl and Open Database Connectivity (ODBC). The "Implementation of DDL Commands" lab experiment focuses on providing students with practical exposure to executing Data Definition Language (DDL). The experiment involves creating database objects, such as tables, views, and indexes, using DDL commands. Through this hands-on experience, students will gain proficiency in effectively utilizing DDL. The lab aims to enhance students' understanding of database management and reinforce their skills in executing DDL.

Pre-Requisites:

PostgreSQL, TerraER Tool, Windows/ Ubuntu/CentOS/Debian, DBMS Concepts.

In Lab Task:

Implementation of basic PostgreSQL Queries with DDL commands.

Schema:

SAILORS (SID: INTEGER, SNAME: STRING, RATING: INTEGER, AGE: REAL);

BOATS (BID: INTEGER, BNAME: STRING, COLOR: STRING);

RESERVES (SID: INTEGER, BID: INTEGER, DAY: DATE).

SAILORS			
SID	SNAME	RATING	AGE
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	56
32	Andy	8	26
58	Rusty	10	35
64	Horatio	7	35
71	Zor	10	16
74	Horatio	9	40
85	Art	3	26
95	Bob	3	64

BOATS		
ID	BNAME	COLOR
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

RESERVES		
SID	BID	DAY
22	101	1998-10-10
22	102	1998-10-10
22	103	1998-10-8
22	104	1998-10-7
31	102	1998-11-10
31	103	1998-11-6
31	104	1998-11-12
64	101	1998-9-5
64	102	1998-9-8
74	103	1998-9-8

In the reserves Questions:

i) Create tables for Sailors, Reserves and Boats with appropriate key constraints?

```
CREATE TABLE Sailors (
    SID INT PRIMARY KEY,
    SNAME VARCHAR(50),
    RATING FLOAT,
    AGE INT
);
```

```
CREATE TABLE Boats (
    ID INT PRIMARY KEY,
    BNAME VARCHAR(50),
    COLOR VARCHAR(20)
);
```

```
CREATE TABLE Reserves (
   SID INT,
   BID INT,
   DAY DATE,
   PRIMARY KEY (SID, BID, DAY),
   FOREIGN KEY (SID) REFERENCES Sailors(SID),
   FOREIGN KEY (BID) REFERENCES Boats(ID)
);
```

ii) Write PostgreSQL Queries to insert values as per the given above Data Base?

```
INSERT INTO Sailors (SID, SNAME, RATING, AGE) VALUES
(22, 'Dustin', 7, 45),
(29, 'Brutus', 1, 33),
(31, 'Lubber', 8, 56),
(32, 'Andy', 8, 26),
(58, 'Rusty', 10, 35),
(64, 'Horatio', 7, 35),
(71, 'Zor', 10, 16),
(74, 'Horatio', 9, 40),
(85, 'Art', 3, 26),
(95, 'Bob', 3, 64);

INSERT INTO Boats (ID, BNAME, COLOR) VALUES
(101, 'Interlake', 'Blue'),
(102, 'Interlake', 'Red'),
(103, 'Clipper', 'Green'),
(104, 'Marine', 'Red');
```

```
INSERT INTO Reserves (SID, BID, DAY) VALUES
(22, 101, '1998-10-10'),
(22, 102, '1998-10-10'),
(22, 103, '1998-10-08'),
(22, 104, '1998-10-07'),
(31, 102, '1998-11-10'),
(31, 103, '1998-11-06'),
(31, 104, '1998-11-12'),
(64, 101, '1998-09-05'),
(64, 102, '1998-09-08'),
(74, 103, '1998-09-08');
```

iii) To retrieve distinct Sailor name and Sailor Age from Sailors?

```
SELECT DISTINCT SNAME, AGE FROM Sailors;
```

iv) To retrieve boat name, boat id of the red and green color boats?

```
SELECT BNAME, ID FROM Boats WHERE COLOR IN ('Red', 'Green');
```

v) Find the names of sailors who have reserved a red boat, and list in the order of age?

```
SELECT DISTINCT S.SNAME, S.AGE
FROM Sailors S
JOIN Reserves R ON S.SID = R.SID
JOIN Boats B ON R.BID = B.ID
WHERE B.COLOR = 'Red'
ORDER BY S.AGE;
```

vi) Find the ids of sailors who have reserved a red boat or a green boat?

```
SELECT DISTINCT R.SID
FROM Reserves R
JOIN Boats B ON R.BID = B.ID
WHERE B.COLOR IN ('Red', 'Green');
```

vii) Find the names of sailors who have reserved at least one boat?

```
SELECT DISTINCT S.SNAME
FROM Sailors S
JOIN Reserves R ON S.SID = R.SID;
```

viii) Display the Insert 'twenty' in age column

```
INSERT INTO Sailors (SID, SNAME, RATING, AGE) VALUES (101, 'John', 5,
'twenty');
```

ix) Display the Insert sid: 100 bid: 205, day: 12-09-17 in reserves table.

```
INSERT INTO Sailors (SID, SNAME, RATING, AGE) VALUES (100, 'NewSailor', 5,
30);

INSERT INTO Boats (ID, BNAME, COLOR) VALUES (205, 'NewBoat', 'Blue');

INSERT INTO Reserves (SID, BID, DAY) VALUES (100, 205, '2017-09-12');
```

x) Create a query that displays the Insert sid: 95, Sname: anil, Rating:3, age: 64 on the sailor's table

```
INSERT INTO Sailors (SID, SNAME, RATING, AGE) VALUES (96, 'Anil', 3, 64);
```

xi) Create a query that displays the update 10.5 as rating where sailor id= 95

```
UPDATE Sailors SET RATING = 10.5 WHERE SID = 95;
```

xii) table give date as per this format like did/mm/yyy

```
SELECT strftime('%d/%m/%Y', DAY) AS FORMATTED_DATE FROM Reserves;
```

Viva-Voce Questions (In-Lab):

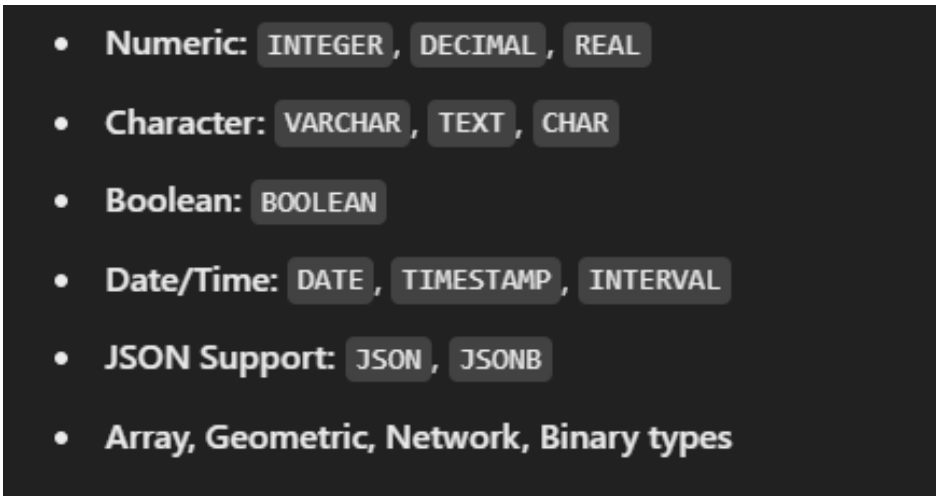
1) What is PostgreSQL?

PostgreSQL is an open-source, object-relational database management system (ORDBMS) known for its extensibility, ACID compliance, and advanced features like JSON support, full-text search, and concurrency control.

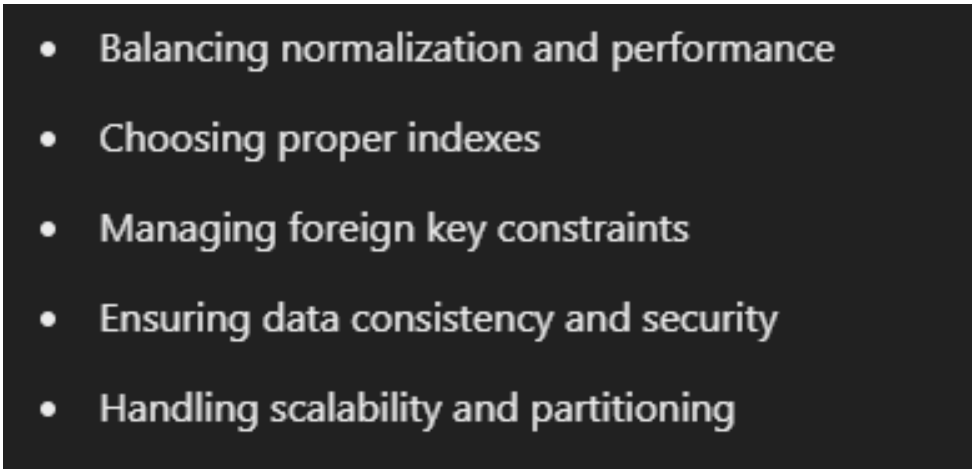
2) What are the advantages of PostgreSQL?

- Open-source and free
- ACID compliance for data integrity
- Supports JSON, arrays, and custom data types
- Advanced indexing and full-text search
- MVCC for high concurrency
- Strong security features

3) Which data types are used in PostgreSQL?



4) What are some common challenges or considerations when designing and implementing complex database schemas using DDL commands?



5) Explain the concept of data integrity constraints in DDL commands and provide examples of commonly used constraints.

Data Integrity Constraints in DDL Commands

Data integrity constraints ensure accuracy and consistency in a database by restricting invalid data entry.
These constraints are applied using **DDL commands** like `CREATE TABLE` and `ALTER TABLE`.

Common Constraints:

1. **NOT NULL** – Prevents `NULL` values.

```sql
Name VARCHAR(50) NOT NULL;
```

2. **UNIQUE** – Ensures all values in a column are unique.

```sql
Email VARCHAR(100) UNIQUE;
```

3. **PRIMARY KEY** – Combines `NOT NULL` and `UNIQUE`.

```sql
StudentID INT PRIMARY KEY;
```

4. **FOREIGN KEY** – Establishes relationships between tables.

```sql
FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID);
```

5. **CHECK** – Restricts values based on a condition.

```sql
Price DECIMAL(10,2) CHECK (Price > 0);
```

6. **DEFAULT** – Assigns a default value.

```sql
Salary DECIMAL(10,2) DEFAULT 50000;
```

Post Lab Task:

Use the tables below and implement the below queries

Customer			
CID	PID	Renatl_Date	Rental_ Period
104	300	10-06-2020	6
105	302	05-07-2020	10
108	304	15-07-2020	3
109	305	25-06-2020	6

Artist			
Aid	Name	Address	Phone
200	John	Delhi	7786549803
201	Samuel	Mumbai	7123458790
202	Samson	Lucknow	9460367777
203	David	Hyderabad	9797276764
204	Raghu	Hyderabad	8134185751
205	Ravi	Mumbai	7471094738
206	Kiran	Delhi	9808003725

Rent			
Pid	Aid	Rental_Cost	Type
300	201	4500	Hired
301	202	3500	Not Hired
302	203	7500	Hired
303	205	2500	Not Hired
304	202	10000	Not Hired
305	201	8000	Not Hired
306	203	6500	Not Hired

Painting				
Cid	Cname	Address	Phone	Category
100	Raju	Hyderabad	9876045789	Bronze
101	Hari	Vijayawada	8877678956	Gold
102	Devi	Guntur	7879312123	Silver
103	Rani	Delhi	8780945290	Platinum
104	Jaya	Mumbai	9612578457	Gold
105	Haritha	Kolkata	9611665513	Silver
106	Kalyan	Vijayawada	9610752569	Bronze
107	Roja	Hyderabad	9609839625	Platinum
108	Amar	Vijayawada	9608926681	Gold
109	Padma	Vijayawada	9608013737	Bronze

Owner				
Pid	Oid	Name	Address	Phone
300	500	Raju	Hyderabad	9460367777
301	500	Hari	Vijayawada	8134185751
302	501	Giri	Hyderabad	7808003725
303	501	Gopi	Delhi	9481821699
304	503	Krishna	Mumbai	7155639673
305	502	Verma	Delhi	8829457647
306	502	Guna	Delhi	7503275621

Create the tables identifying the constraints and relationships.

i) Insert atleast10 records into these tables

```
CREATE TABLE Artist (
   Aid INT PRIMARY KEY,
   Name VARCHAR(50),
   Address VARCHAR(50),
   Phone VARCHAR(15)
);

CREATE TABLE Painting (
   Cid INT PRIMARY KEY,
   Cname VARCHAR(50),
   Address VARCHAR(50),
   Phone VARCHAR(15),
   Category VARCHAR(20)
);

CREATE TABLE Owner (
   Pid INT PRIMARY KEY,
   Oid INT,
   Name VARCHAR(50),
   Address VARCHAR(50),
   Phone VARCHAR(15)
);

CREATE TABLE Customer (
   Cid INT,
   Pid INT,
   Rental_Date DATE,
   Rental_Period INT,
   PRIMARY KEY (Cid, Pid),
   FOREIGN KEY (Pid) REFERENCES Owner(Pid)   );
```

```
CREATE TABLE Rent (
   Pid INT PRIMARY KEY,
   Aid INT,
   Rental_Cost INT,
   Type VARCHAR(20) CHECK (Type IN ('Hired', 'Not Hired')),
   FOREIGN KEY (Aid) REFERENCES Artist(Aid),
   FOREIGN KEY (Pid) REFERENCES Owner(Pid)
);

INSERT INTO Artist VALUES
(200, 'John', 'Delhi', '7786549803'),
(201, 'Samuel', 'Mumbai', '7123458790'),
(202, 'Samson', 'Lucknow', '9460367777'),
(203, 'David', 'Hyderabad', '979727674'),
(204, 'Raghu', 'Hyderabad', '8134185751'),
(205, 'Ravi', 'Mumbai', '7471094738'),
(206, 'Kiran', 'Delhi', '9808003725');


INSERT INTO Painting VALUES
(100, 'Raju', 'Hyderabad', '9876045789', 'Bronze'),
(101, 'Hari', 'Vijayawada', '8876789586', 'Gold'),
(102, 'Devi', 'Guntur', '7879312123', 'Silver'),
(103, 'Rani', 'Delhi', '8780945290', 'Platinum'),
(104, 'Jaya', 'Mumbai', '9612875544', 'Gold'),
(105, 'Haritha', 'Kolkata', '9611665315', 'Silver'),
(106, 'Kalyan', 'Vijayawada', '9610752569', 'Bronze'),
(107, 'Roja', 'Hyderabad', '9609839265', 'Platinum'),
(108, 'Amar', 'Vijayawada', '9609826681', 'Gold'),
(109, 'Padma', 'Vijayawada', '9608013737', 'Bronze');
```

INSERT INTO Owner VALUES

(300, 500, 'Raju', 'Hyderabad', '9460367777'),

(301, 501, 'Hari', 'Vijayawada', '8134185751'),

(302, 502, 'Giri', 'Hyderabad', '7808003725'),

(303, 501, 'Gopi', 'Delhi', '9481821699'),

(304, 502, 'Krishna', 'Mumbai', '7155639673'),

(305, 500, 'Verma', 'Delhi', '8829457647'),

(306, 502, 'Guna', 'Delhi', '7503275621');

INSERT INTO Customer VALUES

(104, 300, '2020-06-10', 6),

(105, 302, '2020-05-07', 10),

(108, 304, '2020-07-15', 3),

(109, 305, '2020-06-25', 6);

INSERT INTO Rent VALUES

(300, 201, 4500, 'Hired'),

(301, 202, 3500, 'Not Hired'),

(302, 203, 7500, 'Hired'),

(303, 205, 2500, 'Not Hired'),

(304, 202, 10000, 'Not Hired'),

(305, 201, 8000, 'Not Hired'),

(306, 203, 6500, 'Not Hired');

ii) Display the customer details who got the category as “Gold”

```
SELECT DISTINCT c.*
FROM Customer c
JOIN Owner o ON c.Pid = o.Pid
JOIN Painting p ON c.CID = p.CID
WHERE p.Category = 'Gold';
```

iii) Display the list of artists who belong to “Hyderabad”

```
SELECT * FROM Artist WHERE Address = 'Hyderabad';
```

iv) Create a PostgreSQL query display the painting details which are not hired

```
SELECT p.*
FROM Painting p
JOIN Owner o ON p.Cname = o.Name
JOIN Rent r ON o.Pid = r.Pid
WHERE r.Type = 'Not Hired';
```

v) Create a PostgreSQL query to update the rental cost of the painting with 1000 for the paintings which are not hired.

```
UPDATE Rent
SET Rental_Cost = 1000
WHERE Type = 'Not Hired';
```

vi) Display the details of the artist whose paintings are hired

```
SELECT DISTINCT a.*
FROM Artist a
JOIN Rent r ON a.Aid = r.Aid
WHERE r.Type = 'Hired';
```

Students Signature

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured:_____ out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation: