NAME-LEPAKSHI JAGMOHAN

BATCH-2

SAP-500119039

DATABASE MANAGEMENT SYSTEMS LAB

EXPERIMENT-8

Title: Use of different SQL clauses and join

Objective: To understand the use of group by and having clause and execute the SQL commands using JOIN

```
create database lab8;
use lab8;
```

1. Consider the following schema:

Student (sid, sname, age)

Match (mid, mname, venue)

Play (sid, mid, day(date))

```
CREATE TABLE Student (
    sid CHAR(10) PRIMARY KEY,
    sname VARCHAR(50),
    age INT
);

CREATE TABLE Matches (
    mid VARCHAR(10) PRIMARY KEY,
    mname VARCHAR(50),
    venue VARCHAR(50)
);

CREATE TABLE Play (
    sid CHAR(10),
    mid CHAR(10),
    day DATE,
    FOREIGN KEY (sid) REFERENCES Student(sid),
    FOREIGN KEY (mid) REFERENCES Matches(mid)
);
```

2. Populate all the tables.

```
INSERT INTO Student (sid, sname, age) VALUES
('S1', 'Amit', 20),
('S2', 'Raj', 22),
('S3', 'Maya', 19),
('S4', 'Sara', 21);

INSERT INTO Matches (mid, mname, venue) VALUES
('B10', 'Cricket', 'Delhi'),
('B11', 'Football', 'Mumbai'),
('B12', 'Tennis', 'Delhi'),
('B13', 'Basketball', 'Kolkata');

INSERT INTO Play (sid, mid, day) VALUES
('S1', 'B10', '2023-11-01'),
('S2', 'B11', '2023-11-01'),
('S3', 'B12', '2023-11-02'),
('S1', 'B11', '2023-11-03'),
('S4', 'B13', '2023-11-04'),
('S2', 'B12', '2023-11-05'),
('S3', 'B10', '2023-11-06');
```

3. Find all information of students who have played match number B10.

```
SELECT *
FROM Student S
JOIN Play P ON S.sid = P.sid
WHERE P.mid = 'B10';
```

4. Find the name of matches played by Amit.

```
SELECT M.mname
FROM Matches M
JOIN Play P ON M.mid = P.mid
JOIN Student S ON P.sid = S.sid
WHERE S.sname = 'Amit';
```

5. Find the names of students who have played a match in Delhi.

```
SELECT DISTINCT S.sname
FROM Student S
JOIN Play P ON S.sid = P.sid
JOIN Matches M ON P.mid = M.mid
WHERE M.venue = 'Delhi';
```

6. Find the names of students who have played at least one match.

```
SELECT DISTINCT S.sname
FROM Student S
JOIN Play P ON S.sid = P.sid;
```

7. Find the ids and names of students who have played two different matches on the same day.

```
SELECT S.sid, S.sname
FROM Student S
JOIN Play P1 ON S.sid = P1.sid
JOIN Play P2 ON S.sid = P2.sid
WHERE P1.day = P2.day AND P1.mid <> P2.mid
GROUP BY S.sid, S.sname;
```

8. Find the ids of students who have played a match in Delhi or Mumbai.

```
SELECT DISTINCT S.sid
FROM Student S
JOIN Play P ON S.sid = P.sid
JOIN Matches M ON P.mid = M.mid
WHERE M.venue IN ('Delhi', 'Mumbai');
```

9. Find the average age of students.

```
SELECT AVG(age) AS Average_Age
FROM Student;
```

NAME-LEPAKSHI JAGMOHAN

BATCH-2

SAPID-500119039

DATABASE MANAGEMENT SYSTEMS LAB

EXPERIMENT-9

Title: To understand the concepts of Views.

Objective: Students will be able to implement the concept of views.

1. Create table of table name: EMPLOYEES and add 6 rows

Column Name	Data Type	Width	Attributes
Employee_id	Character	10	PK
First_Name	Character	30	NN
Last_Name	Character	30	NN
DOB	Date		
Salary	Number	25	NN
Department_id	Character	10	

```
CREATE TABLE EMPLOYEES (
    Employee_id CHAR(10) PRIMARY KEY,
    First_Name VARCHAR(30) NOT NULL,
    Last_Name VARCHAR(30) NOT NULL,
    DOB DATE,
    Salary DECIMAL(10, 2) NOT NULL,
    Department_id CHAR(10)
);
```

```
INSERT INTO EMPLOYEES (Employee_id, First_Name, Last_Name, DOB, Salary, Department_id) VALUES
('E1', 'John', 'Doe', '1980-01-15', 50000, '10'),
('E2', 'Jane', 'Smith', '1985-02-20', 60000, '20'),
('E3', 'Robert', 'Brown', '1978-03-10', 55000, '20'),
('E4', 'Michael', 'Clark', '1990-04-22', 62000, '30'),
('E5', 'Amit', 'Sharma', '1992-06-12', 48000, '10'),
('E6', 'Emma', 'Wilson', '1989-07-14', 67000, '20');
```

2. Execute the following view related queries:

1. Create View of name emp_view and the column would be Employee_id, Last_Name, salary and department_id only.

```
CREATE VIEW emp_view AS
SELECT Employee_id, Last_Name, Salary, Department_id
FROM EMPLOYEES;
```

2. Insert values into view(remove the NOT NULL constraint and then insert values):

```
ALTER TABLE EMPLOYEES MODIFY Last_Name VARCHAR(30);
INSERT INTO emp_view (Employee_id, Last_Name, Salary, Department_id)
VALUES ('E7', 'Green', 53000, '10');
```

3. Modify, delete and drop operations are performed on view.

```
UPDATE emp_view
SET Salary = 55000
WHERE Employee_id = 'E7';

DELETE FROM emp_view WHERE Employee_id = 'E7';

DROP VIEW emp_view;
```

Creates a view named salary_view. The view shows the employees in department 20 and their annual salary.

```
CREATE VIEW salary_view AS
SELECT Employee_id, First_Name, Last_Name, (Salary * 12) AS Annual_Salary
FROM EMPLOYEES
WHERE Department_id = '20';
```