

Database and Management System Lab

Lab Experiment – 8

Name: Arsh Dhama

Batch 2

Sap_ID: 500119526

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

1. Consider the following schema:

Student (sid, sname, age)

Match (mid, mname, venue)

Play (sid, mid, day(date))

```
CREATE TABLE Student (  
    sid INT PRIMARY KEY,  
    sname VARCHAR(50),  
    age INT  
);  
  
CREATE TABLE Match (  
    mid VARCHAR(10) PRIMARY KEY,  
    mname VARCHAR(50),  
    venue VARCHAR(50)  
);  
  
CREATE TABLE Play (  
    sid INT,  
    mid VARCHAR(10),  
    day DATE,  
    PRIMARY KEY (sid, mid, day),  
    FOREIGN KEY (sid) REFERENCES Student(sid),  
    FOREIGN KEY (mid) REFERENCES Match(mid)  
);
```

2. Populate all the tables.

```
INSERT INTO Student (sid, sname, age) VALUES
(1, 'Amit', 20),
(2, 'Rahul', 21),
(3, 'Priya', 22);
```

```
INSERT INTO Match (mid, mname, venue) VALUES
('B10', 'Cricket', 'Delhi'),
('B20', 'Football', 'Mumbai'),
('B30', 'Basketball', 'Delhi');
```

```
INSERT INTO Play (sid, mid, day) VALUES
(1, 'B10', '2024-11-01'),
(1, 'B20', '2024-11-02'),
(2, 'B10', '2024-11-01'),
(3, 'B30', '2024-11-03');
```

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

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

sid	sname	age
1	Amit	20
2	Rahul	21

4. Find the name of matches played by Amit.

```
SELECT m.mname
FROM Student s
JOIN Play p ON s.sid = p.sid
JOIN Match m ON p.mid = m.mid
WHERE s.sname = 'Amit';
```

mname
Cricket
Football

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 Match m ON p.mid = m.mid
WHERE m.venue = 'Delhi';
```

sname
Amit
Rahul
Priya

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;
```

sname
Amit
Rahul
Priya

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.mid <> p2.mid AND p1.day = p2.day
GROUP BY s.sid, s.sname;
```

SQL query successfully executed. However, the result set is empty.

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 Match m ON p.mid = m.mid
WHERE m.venue = 'Delhi' OR m.venue = 'Mumbai';
```

sid
1
2
3

9. Find the average age of students.

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

average_age
21

Database and Management System Lab

Lab Experiment – 9

Name: Garvit Rana

Batch 1

Roll No: R21442230026

Sap_ID: 500119405

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

```
CREATE TABLE EMPLOYEE (  
    Employee_id CHAR(10) PRIMARY KEY,  
    First_Name CHAR(30) NOT NULL,  
    Last_Name CHAR(30) NOT NULL,  
    DOB DATE,  
    Salary NUMERIC(25) NOT NULL  
);
```

2. Create a View emp_view with Specific Columns

```
CREATE VIEW emp_view AS  
SELECT Employee_id, Last_Name, Salary, department_id  
FROM EMPLOYEE;
```

3. Insert Values into the View (emp_view)

```
INSERT INTO EMPLOYEE (Employee_id, First_Name, Last_Name, DOB, Salary, department_id)  
VALUES ('E001', 'John', 'Doe', '1990-05-21', 45000, 'D10');
```

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

```
UPDATE emp_view  
SET Salary = 55000  
WHERE Employee_id = 'E001';
```

```
DELETE FROM emp_view  
WHERE Employee_id = 'E001';
```

```
DROP VIEW emp_view;
```

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

0