## LAB NO 3

## JOIN OPERATIONS IN SQL

#### Objective:

- To learn and apply different types of Join Operations in SQL.
- To understand and implement foreign key.

#### LAB TASKS

1. Consider the following schema:

Sailors (sid(PK), sname, rating, age) Boats (bid(PK), bname, color) Reserves (sid (FK), bid(FK), day(date)).

Find all information of sailors who have reserved boat number 101.

## Input 01:

```
create database lab3;
      use lab3;
 3
 4 • create table Sailors (sid int primary key, sname varchar(25), rating varchar(25), age int);
 5 • insert into Sailors (sid, sname, rating, age) values
    (101, 'WAQAR RIASAT ALI', '9.9%', 20),
    (230, 'ANEEQ SHAMS MACHERA', '8.9%', 27),
 7
     (216, 'HUSSAIN MACHERA', '7.9%', 26),
    (219, 'QASIM MACHERA', '6.9%', 25),
      (250, 'SAQIB MACHERA', '5.9%', 24),
10
      (376, 'HASSAN MACHERA', '4.9%', 23),
11
      (220, 'ADEEL MACHERA', '3.9%', 22);
12
13
14
15 • create table Boats(B_id int primary key, bname varchar(25), color varchar(25));
16 • insert into Boats (B_id , bname, color) values
17
     (101, 'Surfing Boat', 'Blue'),
      (230, 'Fishing Boat', 'Red'),
18
      (216, 'Speed Boat', 'White'),
19
      (219, 'Sail Boat', 'Green'),
```

```
🚞 🖫 | 🗲 🖟 👰 🔘 | 🚳 | 💿 🔞 | Limit to 1000 rows 🔻 | 🛵 | 🥩 🔍 🗻
      insert into Boats (B_id , bname, color) values
      (101, 'Surfing Boat', 'Blue'),
     (230, 'Fishing Boat', 'Red'),
     (216, 'Speed Boat', 'White'),
     (219, 'Sail Boat', 'Green'),
21
     (250, 'Rescue Boat', 'Yellow'),
      (376, 'Rowing Boat', 'Brown'),
22
      (220, 'Luxury Yacht', 'Black');
23
24
25 \ominus create table Reserves(sid INT, B_id INT ,day date , foreign key (sid) references Sailors(sid), foreign key (B_id)
26
      references Boats(B_id) );
27
28
      INSERT INTO Reserves (sid, B_id, day) VALUES
      (101, 101, '2025-04-18'),
29
       (230, 101, '2025-04-19');
31
32
33 • SELECT S.sid, S.sname, B.bname, R.day -- sailors_idi ,sailors_name & Boats_idi, Boats_name & Reserve_day choose kro.
                                              -- 'S' is the short form of saiors used above
34
      JOIN Reserves R ON S.sid = R.sid -- agar sailors table me se koyi bhi idi match hojaye reserves table ki idi se or
35
     JOIN Boats B ON R.B_id = B.B_id -- agar Boats table me se koyi bhi idi match hojaye reserves table ki Boat idi se to
36
                                  -- condition of Boat idi "101" to "S.sid, S.sname, B.bname, R.day" show krdo
37
      WHERE R.B id = 101:
```

# Output 01:

	sid	sname	bname	day
•	101	WAQAR RIASAT ALI	Surfing Boat	2025-04-18
	230	ANEEQ SHAMS MACHERA	Surfing Boat	2025-04-19

**2.** Write a query to find and display the name of boat reserved by Bob

## Input 02:

```
45
46 • SELECT S.sname, B.bname, R.day
47 FROM Sailors S
48 JOIN Reserves R ON S.sid = R.sid
49 JOIN Boats B ON R.B_id = B.B_id -
50 WHERE R.sid = 101;
```

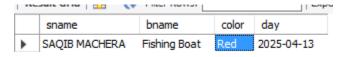
## Output 02:

	sname	bname	day
•	WAQAR RIASAT ALI	Surfing Boat	2025-04-18
	WAQAR RIASAT ALI	Surfing Boat	2025-04-18
	WAQAR RIASAT ALI	Rescue Boat	2025-04-01
	WAQAR RIASAT ALI	Sail Boat	2025-04-01

3. Find the names of sailors who have reserved a red boat, and list in the order of age.

## Input 03:

#### Output 03:

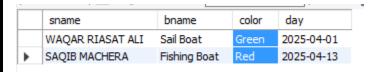


4. Find the ids of sailors who have reserved a red boat or a green boat.

## Input 04:

```
54     SELECT S.sname, B.bname, B.color, R.day
55     FROM Sailors S
56     JOIN Reserves R ON S.sid = R.sid
57     JOIN Boats B ON R.B_id = B.B_id
58     WHERE B.color = 'red' OR B.color = 'Green';
```

## Output 04:

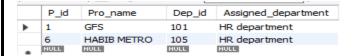


5. Create a table Projects with the following schema. (P\_id, Pro\_name, Dep\_id). Now write a query to show which projects have been assigned to the 'HR' department. 6

## Input 05:

```
-- TASK NO: 5
65
66 •
       create table Projects(P_id int Primary key , Pro_name varchar(25), Dep_id int, Assigned_department varchar(20));
67
       INSERT INTO Projects (P_id, Pro_name, Dep_id, Assigned_department) VALUES
68 •
69
       (1, 'GFS', 101, 'HR department'),
70
       (2, 'ARY LAGUNA', 102, 'Finance department'),
       (3, 'BAHRIA TOWN KARACHI', 103, 'Sales department'),
71
       (5, 'DHA ISLAMABAD', 104, 'Marketing department'),
72
       (6, 'HABIB METRO', 105, 'HR department'),
73
       (7, 'KDA LEASE', 106, 'Audit department'),
75
       (8, 'SCHEME33', 107, 'Technical department');
76
77 •
       select * from Projects
78
       where Assigned_department = 'HR department' ;
```

## Output 05:

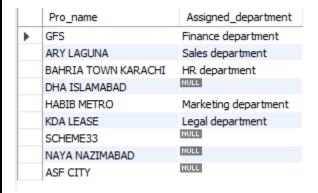


6. Write a query to display project names along with their allocated department names. Also display the names of those projects which have not been assigned to any department yet.

#### Input 06:

```
19 • ⊖ create table department (
           Assigned_department varchar(25),
20
           P id int,
21
22
           foreign key(P_id) references Projects(P_id));
23
       insert into department (Assigned department, P id) values
24 •
       ('Finance department', 1),
25
       ('Sales department', 2),
26
       ('HR department', 3),
27
28
       (null, 5),
       ('Marketing department', 6),
29
       ('Legal department', 7),
30
       (null, 9);
31
32
33 •
       select Projects.Pro_name, department.Assigned_department
34
       from Projects
       Left join
35
36
       department
37
38
       Projects.P_id = department.P_id;
```

## Output 06:



7. Create tables for the student, teacher and course scenario as shown in the manual. Write a query to display the course name along with the allocated teacher's name for that course.

## Input 07

## Output 07:

