

Task 5.1 - Study of SQL Joins

1. Inner join - return only matching rows from both tables.

```
SELECT e.EmpName, e.Dept,  
d.ManagerName  
FROM Employee e  
INNER JOIN Department d ON e.Dept = d.Dept;
```

2. Left join (or Left outer join) - returns all rows from the left table, and matching rows from the right table (NULL if no match).

```
SELECT e.EmpName, e.Dept, n.city  
FROM Employee e  
LEFT JOIN NewEmployee n ON e.EmpID = n.EmpID;
```

3. Right join (or right outer join) - returns all rows from the right table, and matching rows from the left.

```
SELECT e.EmpName, n.city  
FROM Employee e  
RIGHT JOIN NewEmployee n ON e.EmpID = n.EmpID;
```

4. Full Outer join - not directly in MySQL, but can be simulated with UNION.

```
SELECT e.EmpName, n.city  
FROM Employee e  
LEFT JOIN NewEmployee n ON e.EmpID = n.EmpID  
UNION  
SELECT e.EmpName, n.city  
FROM Employee e
```

RIGHT JOIN

5. Cross join :- cartesian product (every row with every row).

```
SELECT e.EmpName, n.EmpName  
FROM Employee e  
CROSS JOIN NewEmployee n;
```

TASKS-2 - performing advanced query processing

1. Find top 3 highest-paid employees

```
SELECT EmpName, salary
FROM Employee
ORDER BY salary DESC
LIMIT 3;
```

2. Find the department with the highest average salary:

```
SELECT Dept
FROM Employee
GROUP BY Dept
ORDER BY AVG(salary) DESC
LIMIT 1;
```

3. Find employees who have the same salary as someone else (duplicate salaries)

```
SELECT EmpName, salary
FROM Employee
WHERE salary IN (
    SELECT salary
    FROM Employee
    GROUP BY salary
    HAVING COUNT(*) > 1)
```

4. Find employees who earn more than all HR emp.

```
SELECT EmpName, salary
FROM Employee
WHERE salary > ALL (SELECT salary FROM Employee
WHERE Dept = 'HR');
```

5. List departments and their highest-paid employee

```
SELECT e.Dept, e.EmpName, e.Salary
FROM Employee e
WHERE salary = (
    SELECT MAX(salary)
    FROM Employee
    WHERE Dept = e.Dept)
```

6. Find employees who joined earliest in each department.

```
SELECT e.Dept, e.EmpName,
e.joiningDate
FROM Employee e
WHERE joiningDate = (
    SELECT MIN(joiningDate)
    FROM Employee
    WHERE Dept = e.Dept);
```

VEL TECH - CSE	
EX NO.	5
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	15
DATE WITH DATE	15/12/25

Result:-

Hence the study of SQL joins are successfully verified and output also verified.

15/12/25