**Day 12: 31-10-2025:**

**This below link contains sample tables**

[**https://github.com/Kaleakash/mysql\_tables.git**](https://github.com/Kaleakash/mysql_tables.git)

Where clause: filter the data

1. Relational operator >, <, >=, <=, =, !=

select \* from employee where salary > 15000;

1. Between operator : range of data

select \* from employee where salary between 5000 and 10000;

select \* from employee where hire\_date between '1995-01-01' and '1998-12-31';

1. In operator : multiple value with condition

select \* from employee where job\_id in('SA\_REP','ST\_CLERK');

1. Like : like equal with special features as start with, end with, contains etc with %

select \* from employee where first\_name ='steven';

select \* from employee where first\_name like 'steven';

% means 0 or 1 or many

Start with s character

select \* from employee where first\_name like 's%';

end with a character

select \* from employee where first\_name like '%a';

contains g character

select \* from employee where first\_name like '%g%';

\_

Only one character anything

select \* from employee where first\_name like 's\_e%';

1. is null

select \* from employee where manager\_id is null;

select \* from employee where manager\_id is not null

**Order by clause**

It is use to sort the data in ascending or descending order

select \* from employee order by salary asc;

select \* from employee order by salary desc;

select \* from employee order by first\_name asc;

select \* from employee order by first\_name asc;

Join:

Join is use to retrieve more than one column from more than one table with or without conditions.

1. Cross join or cartesian product

M \* N

select first\_name,salary,department\_name from employee, department;

select first\_name,salary,department\_id,department\_name from employee, department;

error

select employee.first\_name,employee.salary,department.department\_id,department.department\_name from employee, department;

**we can table alias**

select emp.first\_name,emp.salary,dept.department\_id,dept.department\_name from employee emp, department dept;

**join types**

1. **Inner join (Equi join) :**

it display common records present in both the tables.

select emp.first\_name,emp.salary,dept.department\_id,dept.department\_name from employee emp inner join department dept on emp.department\_id=dept.department\_id;

1. Outer join
   1. Left outer join

Common as well as left table remaining record

**select emp.first\_name,emp.salary,dept.department\_id,dept.department\_name from employee emp left outer join department dept on emp.department\_id=dept.department\_id;**

* 1. Right outer join

Common as well as right table remaining record

**select emp.first\_name,emp.salary,dept.department\_id,dept.department\_name from employee emp right outer join department dept on emp.department\_id=dept.department\_id;**

* 1. Full join

Common + left and right remaining records