**Day 8: 2 March 2025**

create database databasename

create database mytestdb; it is use to create new database.

use mytestdb; now we are inside a new database.

show tables;

connecting mysql database using workbench (GUI client application)

**Join**

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

1. Cross join : m\*n =

select \* from users;

select \* from user\_profile;

select email,password,first\_name,last\_name,age

from users,user\_profile;

1. Inner join or equi join :

We write condition with common columns.

select email,password,first\_name,last\_name,age from users inner join user\_profile on uid=profile\_id;

**inner join with table alias**

select e.first\_name,e.email,d.department\_id,d.department\_name,d.location from employees e inner join department d on e.department\_id=d.department\_id;

1. Left outer join

In left outer join common as well left side or first table remaining record it will display.

1. Right outer join

In right outer join common as well right side or second table remaining record it will display.

1. Full outer join : common + left side remaining record as well as right side remaining records.

select e.first\_name,e.email,d.department\_id,d.department\_name,d.location

from employees e left outer join department d on e.department\_id=d.department\_id

union

select e.first\_name,e.email,d.department\_id,d.department\_name,d.location

from employees e right outer join department d on e.department\_id=d.department\_id;

**functions**

function contains set of instruction to perform a specific task.

Functions mainly divided into 2 types.

1. Pre defined functions : part of SQL
2. User defined or custom function : part of PL SQL

Pre defined functions

Divided into two types.

1. Single row functions

The function functionality apply for each records individually.

1. Multi row functions

The function functionality apply for more than one records base upon groups.

Single row functions

String functions.

upper(‘value’)

select upper('steven');

select upper('steven') as Names;

select first\_name,upper(first\_name) as NamesInUpperCase from employees;

select first\_name,lower(first\_name) as LowerCaseName from employees;

select concat(first\_name,last\_name) as FullName from employees;

select concat(first\_name,' ',last\_name) as FullName from employees;

select length(email) as NumberOfEmailIdCharacters from employees;

select substring(first\_name,3) from employees;

select substring(first\_name,3,2) from employees;

**number function**

select round(456.456789,2); 456.46

select round(456.456789,1); 456.5

select truncate(456.456789,2); 456.45

select truncate(456.456789,1); 456.4

select ceil(10.9); 11

select ceil(10.1); 11

select floor(10.1); 10

select floor(10.9); 10

**date function**

select curdate(); display current date

select curtime(); display current time

select now(); display current date and time.

date\_add(now(), interval value unit)

select date\_add(now(),interval 10 day);

select date\_add(now(),interval 5 month);

select date\_add(now(),interval 2 year);

select date\_sub(now(),interval 2 year);

select first\_name,doj,datediff(now(),doj)/356 from employees;

Multi row or aggregate functions

Sum()

Max()

Min()

Avg()

Count() : count ignore the null value.

select sum(salary) totalSalary from employees;

select max(salary) maxSalary from employees;

select min(salary) minSalary from employees;

select avg(salary) avgSalary from employees;

select count(employee\_id) from employees;

select count(\*) from employees;

whenever we use aggregate function by default whole table consider as one group. But if we want to use sub group. Then we need to use group by clause.

select department\_id,sum(salary) totalSalary from employees group by department\_id;

select department\_id,count(\*) totalEmployeeWorking from employees group by department\_id;

group by clause : group by clause we use with aggregate or multi row functions.

having clause : this clause we use after group by. It is like a where clause. Where clause must be before group by which help to apply the condition for each records. Having clause like a where clause we apply condition on group of record or result.

Group by with department\_id with where clause

select department\_id,sum(salary) totalSalary from employees where department\_id is not null group by department\_id;

Group by with department\_id with where clause as well as having clause

select department\_id,sum(salary) totalSalary from employees where department\_id is not null group by department\_id having sum(salary) > 70000;

order by clause: this clause is use to do sorting

select \* from employees order by age asc;

select \* from employees order by age desc;

select \* from employees order by salary asc;

select \* from employees order by salary asc;

select \* from employees order by salary; by default ascending order consider

sub query :

sub query : query within another query.

Find the employee name whose salary is Grether average salary of all employee or working in particular department.

Syntax

Outer query (inner Query)

Outer query (Inner Query (Inner Inner Query))

First inner query execute. The output of inner query is input for outer query.

Generally in inner query we need to use only one column. But base upon inner query few query return one result and few query return many records or result.

If inner query use aggregate function without group by it return only single result.

If inner query use where clause with pk it can return single result.

Single row sub query

Outer Query RO >,>=,<,<=,=,!= (Inner Query)

select first\_name,salary from employees where salary > (select avg(salary) from employees);

Multi row sub query

Outer query in /Any with RO/All with RO (Inner Query)

In operator

Find employee details or name who belongs to the department that have more than 1 employees are working.

select first\_name from employees where department\_id in(select department\_id from employees group by department\_id having count(\*) >=2);

select salary from employees where department\_id=2

this query return more than one records. Min salary 2700 and max 32000

find the employee details whose salary is greater any of the employee working in department 2.

select first\_name,salary from employees where salary >any (select salary from employees where department\_id=2);

find the employee details whose salary is greater all of the employee working in department 2.

select first\_name,salary from employees where salary >all (select salary from employees where department\_id=2);