**Barath S Sundar**

**Impact Training – Trainee**

**SQL and PL/SQL Queries**

**SQL Questions**

**Basic Queries**

1. Write a query to display all rows and columns from the employees table.

select\*from employee;

2. Retrieve only the name and salary of all employees from the employees table.

select name,salary from employee;

3. Write a query to find all employees whose salary is greater than 50,000.

select\*from employee where salary>30000;

4. List all employees who joined the company in the year 2020.

select\*from employee where year(doj)=2024;

5. Retrieve the details of employees whose names start with the letter 'A'.

select\*from employee where name like 'c%';

**Aggregate Functions**

1. Write a query to calculate the average salary of all employees.

select avg(salary) as average\_salary\_of\_employees from employee;

2. Find the total number of employees in the company.

select count(\*) as total\_no\_of\_employee from employee;

3. Write a query to find the highest salary in the employees table.

select max(salary) as highest\_salary from employee;

4. Calculate the total salary paid by the company for all employees.

select sum(salary) as total\_salary from employee;

5. Find the count of employees in each department.

select count(\*) from employee group by dept;

**Joins**

1. Write a query to retrieve employee names along with their department names (using employees and departments tables).

2. List all employees who have a manager (self-join on employees table).

3. Find the names of employees who are working on multiple projects (using employees and projects tables).

4. Write a query to display all projects and the employees assigned to them.

5. Retrieve the names of employees who do not belong to any department.

**Subqueries**

1. Write a query to find the employees with the second-highest salary.

select\*from employee order by salary desc limit 1 offset 1;

2. Retrieve the names of employees whose salary is above the department average salary.

select name from employee where salary > (select avg(salary) from employee);

3. Find employees who earn more than the average salary of the entire company.

select dept,count(\*) as num\_of\_employee from employee group by dept;

4. Write a query to find the department with the highest number of employees.

select \* from employee where city='chennai';

5. List all employees who work in a department located in 'New York'.

**Set Operators**

1. Write a query to find employees who work in either the 'HR' or 'Finance' department.

select name from employee where dept='HR' or dept='ES';

2. Retrieve the names of employees who are working on both Project A and Project B.

select name from employee where project = 'A' and project = 'B';

3. Find employees who are not assigned to any project.

select \* from employee where projectId is null;

4. Write a query to get all unique job titles across all departments.

select distinct dept from employee;

5. Combine two tables (employees and former\_employees) and remove duplicates.

select\*from employee union select\*from former\_employee;

**DML and DDL**

1. Write a query to add a new employee to the employees table.

insert into employee values(12305,'Ravi','HR','Pudukottai',45000,'2001-12-01');

2. Update the salary of all employees in the 'IT' department by 10%.

update employee set salary = salary\*1.10 where dept='Insurance';

3. Delete all employees who have not worked for more than 5 years.

delete from employee where year(curdate())-year(doj) <=5;

4. Create a new table departments\_backup with the same structure as the departments table.

create table department(D\_Id int primary key,Dept\_name varchar(20));

5. Drop the temporary\_data table from the database.

drop table department;

**Constraints**

1. Add a primary key to the employees table.

Alter table employee add primary key(EmpId);

2. Write a query to create a foreign key between employees and departments tables.

Alter table employee add foreign key(dept\_id) references department(D\_Id);

3. Add a unique constraint to the email column in the employees table.

Alter table employee add unique (email);

4. Write a query to check all constraints applied on the employees table.

show create table employee;

5. Remove the NOT NULL constraint from the phone\_number column in the employees table.

Alter table employee modify salary int null;