

Name: Sharan S

Practice: Impact Training

Designation: Trainee

SQL Queries

Basic Queries

1. Write a query to display all rows and columns from the employees table.
2. Retrieve only the name and salary of all employees from the employees table.
3. Write a query to find all employees whose salary is greater than 50,000.
4. List all employees who joined the company in the year 2020.
5. Retrieve the details of employees whose names start with the letter 'A'.

Answers:

- `select*from employee;`
- `select name,salary from employee;`
- `select*from employee where salary>30000;`
- `select*from employee where year(doj)=2024;`
- `select*from employee where name like 'c%';`

Aggregate Functions

1. Write a query to calculate the average salary of all employees.
2. Find the total number of employees in the company.
3. Write a query to find the highest salary in the employees table.

4. Calculate the total salary paid by the company for all employees.
5. Find the count of employees in each department.

Answers:

- select avg(salary) as average_salary_of_employees from employee;
- select count(*) as total_no_of_employee from employee;
- select max(salary) as highest_salary from employee;
- select sum(salary) as total_salary from employee;
- 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.

- select e.name, d.dep_name from employees e

join departments d on e.dep_id = d.dep_id;

- select distinct e.name,e1.manager_id from employees e join employees e1

on e1.manager_id = e.manager_id;

- select e.name,p.proj_id from employees e

join projects p on e.emp_id = e.emp_id group by(e.emp_id)

having count(distinct p.proj_id)>1 ;

- select e.name,p.proj_name from projects p

left join employees e on e.proj_id=p.proj_id;

- select e.name from employees e

left join departments d on e.dep_id = d.dep_id

where d.dep_id is null;

Subqueries

1. Write a query to find the employees with the second-highest salary.
2. Retrieve the names of employees whose salary is above the department average salary.
3. Find employees who earn more than the average salary of the entire company.
4. Write a query to find the department with the highest number of employees.
5. List all employees who work in a department located in 'New York'.

Answers:

- `select * from employee order by salary desc limit 1 offset 1;`
- `select name from employee where salary > (select avg(salary) from employee);`
- `select dept, count(*) as num_of_employee from employee group by dept;`
- `select * from employee where city='chennai';`

Set Operators

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

2. Retrieve the names of employees who are working on both Project A and Project B.
3. Find employees who are not assigned to any project.
4. Write a query to get all unique job titles across all departments.
5. Combine two tables (employees and former_employees) and remove duplicates.

Answers:

- select name from employee where dept='HR' or dept='ES';
- select name from employee where project = 'A' and project = 'B';
- select * from employee where projectId is null;
- select distinct dept from employee;
- select*from employee union select*from former_employee;

DML and DDL

1. Write a query to add a new employee to the employees table.
2. Update the salary of all employees in the 'IT' department by 10%.
3. Delete all employees who have not worked for more than 5 years.
4. Create a new table departments_backup with the same structure as the departments table.

5. Drop the temporary_data table from the database.

Answers:

- insert into employee values(12305,'Ravi','HR','Pudukottai',45000,'2001-12-01');
- update employee set salary = salary*1.10 where dept='Insurance';
- delete from employee where year(curdate())-year(doj) <=5;
- create table department(D_Id int primary key,Dept_name varchar(20));
- drop table department;

Constraints

1. Add a primary key to the employees table.
2. Write a query to create a foreign key between employees and departments tables.
3. Add a unique constraint to the email column in the employees table.
4. Write a query to check all constraints applied on the employees table.
5. Remove the NOT NULL constraint from the phone_number column in the employees table.

Answers:

- Alter table employee add primary key(EmpId);
- Alter table employee add foreign key(dept_id) references department(D_Id);
- Alter table employee add unique (email);
- show create table employee;
- Alter table employee modify salary int null;