1. Difference between INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN with example.

Inner join shows only the similar rows between two tables, if rows are not similar then they deleted.

Ex. SELECT e.emp\_name, d.dept\_name

FROM Employee e

INNER JOIN Department d ON e.dept\_id = d.dept\_id;

Left join returns **all rows from the left table** and matching rows from the right table. If there is no match then null values in right table.

Ex. SELECT Customers.CustomerName, Orders.OrderID

FROM Customers

LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID;

Right join returns **all rows from the right table** and matching rows from the leftt table. If there is no match then null values in leftt table.

Ex. SELECT e.emp\_name, d.dept\_name

FROM Employee e

RIGHT JOIN Department d ON e.dept\_id = d.dept\_id;

1. How to find the second highest salary in a table?
2. Using MAX subquery

SELECT MAX(salary) AS second\_highest\_salary

FROM Employee

WHERE salary < (SELECT MAX(salary) FROM Employee);

1. Using ORDER BY with LIMIT

SELECT salary

FROM Employee

ORDER BY salary DESC

LIMIT 1 OFFSET 1;

1. What are Aggregate functions in SQL? Explain with example.

Aggregate functions in SQL are MAX(), MIN(),AVG(),SUM(),COUNT().

EX.1) SELECT MAX(salary) AS highest\_salary,

MIN(salary) AS lowest\_salary

FROM Employee;

2) SELECT AVG(salary) AS average\_salary

FROM Employee;

3) SELECT SUM(salary) AS total\_salary

FROM Employee;

4) SELECT COUNT(\*) AS total\_employees

FROM Employee;

5.Write a query to find duplicate records in a table.

SELECT emp\_name, COUNT(\*) AS duplicate\_count

FROM Employee

GROUP BY emp\_name

HAVING COUNT(\*) > 1;

6. Explain Primary Key, Foreign Key, Unique Key differences.

A primary key uniquely identifies each row in a table. Foreign key create relation between two table and Unique key ensure that all values in column are unique.

Q1.Ans.

SELECT e.emp\_name AS Employee, d.dept\_name AS Department

FROM Employee e

JOIN Department d ON e.dept\_id = d.dept\_id;

Q2.Ans.

SELECT

e.emp\_name AS Employee,

e.job\_title,

e.salary,

e.hire\_date,

p.gender,

p.dob,

p.phone,

p.email,

p.city,

p.marital\_status

FROM Employee e

JOIN Emp\_Personal\_Detail p ON e.emp\_id = p.emp\_id;

Q3. Find employees whose salary is greater than average salary.

SELECT emp\_name, salary

FROM Employee

WHERE salary > (SELECT AVG(salary) FROM Employee);

Q4. Show salary category using CASE (High/Low).

SELECT

emp\_name,

salary,

CASE

WHEN salary >= (SELECT AVG(salary) FROM Employee) THEN 'High'

ELSE 'Low'

END AS salary\_category

FROM Employee;

Q5. combine department names and employee names.

SELECT CONCAT(e.emp\_name, ' - ', d.dept\_name) AS Employee\_Department

FROM Employee e

JOIN Department d ON e.dept\_id = d.dept\_id;

Q6. Find employees whose email ends with @email.com (REGEXP).

SELECT e.emp\_name, p.email

FROM Employee e

JOIN Emp\_Personal\_Detail p ON e.emp\_id = p.emp\_id

WHERE p.email REGEXP '@email\\.com$';

Q7. Count employees in each department.

SELECT d.dept\_name, COUNT(e.emp\_id) AS total\_employees

FROM Department d

LEFT JOIN Employee e ON d.dept\_id = e.dept\_id

GROUP BY d.dept\_name;

Q8. Find the second highest salary employee.

SELECT emp\_name, salary

FROM Employee

WHERE salary = (

SELECT MAX(salary)

FROM Employee

WHERE salary < (SELECT MAX(salary) FROM Employee)

);