Here are **50 commonly asked SQL queries** and use cases. These cover basics, joins, subqueries, aggregate functions, date functions, and more.

**✅ 1–10: Basic SELECT Queries**

1. SELECT \* FROM Employees;
2. SELECT FirstName, LastName FROM Employees;
3. SELECT DISTINCT Department FROM Employees;
4. SELECT \* FROM Employees WHERE Department = 'HR';
5. SELECT \* FROM Employees WHERE Salary > 50000;
6. SELECT \* FROM Employees WHERE HireDate BETWEEN '2020-01-01' AND '2023-01-01';
7. SELECT \* FROM Employees WHERE Name LIKE 'A%';
8. SELECT \* FROM Employees WHERE Department IN ('HR', 'Finance');
9. SELECT \* FROM Employees WHERE ManagerID IS NULL;
10. SELECT TOP 5 \* FROM Employees ORDER BY Salary DESC;

**✅ 11–20: Aggregate Functions**

1. SELECT COUNT(\*) FROM Employees;
2. SELECT MAX(Salary) FROM Employees;
3. SELECT MIN(Salary) FROM Employees;
4. SELECT AVG(Salary) FROM Employees;
5. SELECT SUM(Salary) FROM Employees;
6. SELECT Department, COUNT(\*) FROM Employees GROUP BY Department;
7. SELECT Department, AVG(Salary) FROM Employees GROUP BY Department;
8. SELECT Department, COUNT(\*) FROM Employees GROUP BY Department HAVING COUNT(\*) > 5;
9. SELECT Gender, COUNT(\*) FROM Employees GROUP BY Gender;
10. SELECT Department, MAX(Salary) FROM Employees GROUP BY Department;

**✅ 21–30: Joins**

1. SELECT e.Name, d.DeptName

FROM Employees e

INNER JOIN Departments d ON e.DeptID = d.ID;

22.SELECT e.Name, d.DeptName

FROM Employees e

LEFT JOIN Departments d ON e.DeptID = d.ID;

1. SELECT e.Name, d.DeptName

FROM Employees e

RIGHT JOIN Departments d ON e.DeptID = d.ID;

1. SELECT e.Name, d.DeptName

FROM Employees e

FULL OUTER JOIN Departments d ON e.DeptID = d.ID;

1. SELECT o.OrderID, c.CustomerName

FROM Orders o

JOIN Customers c ON o.CustomerID = c.CustomerID;

1. SELECT o.OrderID, p.ProductName

FROM Orders o

JOIN OrderDetails od ON o.OrderID = od.OrderID

JOIN Products p ON od.ProductID = p.ProductID;

1. SELECT e.Name, m.Name AS Manager

FROM Employees e

JOIN Employees m ON e.ManagerID = m.ID;

1. SELECT \*

FROM Employees e

CROSS JOIN Departments d;

1. SELECT d.DeptName, COUNT(e.ID)

FROM Departments d

LEFT JOIN Employees e ON d.ID = e.DeptID

GROUP BY d.DeptName;

1. SELECT \*

FROM Employees

JOIN Salaries ON Employees.ID = Salaries.EmpID

WHERE Salaries.Amount > 50000;

**✅ 31–40: Subqueries**

1. SELECT Name, Salary

FROM Employees

WHERE Salary > (SELECT AVG(Salary) FROM Employees);

1. SELECT Name

FROM Employees

WHERE DeptID IN (SELECT ID FROM Departments WHERE Location = 'NY');

1. SELECT Name

FROM Employees

WHERE EXISTS (SELECT \* FROM Projects WHERE Projects.EmpID = Employees.ID);

1. SELECT Name

FROM Employees

WHERE ID = (SELECT MAX(ID) FROM Employees);

1. SELECT Name

FROM Employees

WHERE Salary = (SELECT MAX(Salary) FROM Employees WHERE DeptID = 1);

1. SELECT \*

FROM Orders

WHERE CustomerID NOT IN (SELECT CustomerID FROM Customers WHERE Country = 'Germany');

1. SELECT Name

FROM Employees

WHERE DeptID = (SELECT ID FROM Departments WHERE DeptName = 'IT');

1. SELECT \*

FROM Orders

WHERE OrderID IN (SELECT OrderID FROM OrderDetails WHERE Quantity > 10);

1. SELECT Name

FROM Employees

WHERE Salary BETWEEN (SELECT MIN(Salary) FROM Employees) AND (SELECT MAX(Salary) FROM Employees);

1. SELECT \*

FROM Products

WHERE Price > ALL (SELECT Price FROM Products WHERE Category = 'Electronics');

**✅ 41–50: Miscellaneous Queries**

1. UPDATE Employees SET Salary = Salary \* 1.1 WHERE Department = 'Sales';
2. DELETE FROM Employees WHERE Resigned = 1;
3. INSERT INTO Departments (DeptName, Location) VALUES ('Marketing', 'Hyderabad');
4. SELECT GETDATE();
5. SELECT DATEDIFF(year, HireDate, GETDATE()) AS YearsOfService FROM Employees;
6. SELECT UPPER(Name) FROM Employees;
7. SELECT LEFT(Name, 3) FROM Employees;
8. SELECT \* FROM Employees ORDER BY Name ASC;
9. SELECT \* FROM Employees WHERE Email IS NOT NULL;
10. SELECT Name + ' - ' + Department AS EmployeeInfo FROM Employees;