## SQL Query Interview Questions with Answers

**Ques.1. Write a SQL query to fetch the count of employees working in project 'P1'.**  
Ans. Here, we would be using aggregate function count() with the SQL where clause-

**SELECT** **COUNT**(\*) **FROM** EmployeeSalary **WHERE** Project = 'P1';

**Ques.2. Write a SQL query to fetch employee names having salary greater than or equal to 5000 and less than or equal 10000.**  
Ans. Here, we will use BETWEEN in the 'where' clause to return the empId of the employees with salary satifying the required criteria and then use it as subquery to find the fullName of the employee form EmployeeDetails table.

**SELECT** FullName

**FROM** EmployeeDetails

**WHERE** EmpId **IN**

(**SELECT** EmpId **FROM** EmpolyeeSalary

**WHERE** Salary **BETWEEN** 5000 **AND** 10000);

**Ques.3. Write a SQL query to fetch project-wise count of employees sorted by project's count in descending order.**  
Ans. The query has two requirements - first to fetch the project-wise count and then to sort the result by that count. For project wise count, we will be using GROUPBY clause and for sorting, we will use ORDER BY clause on the alias of the project-count.

**SELECT** Project, **count**(EmpId) EmpProjectCount

**FROM** EmployeeSalary

**GROUP** **BY** Project

**ORDER** **BY** EmpProjectCount **DESC**;

**Ques.4. Write a query to fetch only the first name(string before space) from the FullName column of EmployeeDetails table.**  
Ans. In this question, we are required to first fetch the location of the space character in the FullName field and then extract the first name out of the FullName field. For finding the location we will use LOCATE method in mySQL and CHARINDEX in SQL SERVER and for fetching the string before space, we will use SUBSTRING OR MID method.

**mySQL- Using MID**

**SELECT** MID(FullName, 0, LOCATE(' ',FullName)) **FROM** EmployeeDetails;

**SQL Server-Using SUBSTRING**

**SELECT** SUBSTRING(FullName, 0, CHARINDEX(' ',FullName)) **FROM** EmployeeDetails;

**Also, we can use LEFT which returns the left part of a string till specified number of characters.**

**SELECT** **LEFT**(FullName, CHARINDEX(' ',FullName) - 1) **FROM** EmployeeDetails;

**Ques.5. Write a query to fetch employee names and salary records. Return employee details even if the salary record is not present for the employee.**  
Ans. Here, we can use left join with EmployeeDetail table on the left side.

**SELECT** E.FullName, S.Salary

**FROM** EmployeeDetails E **LEFT** **JOIN** EmployeeSalary S

**ON** E.EmpId = S.EmpId;

**Ques.6. Write a SQL query to fetch all the Employees who are also managers from EmployeeDetails table.**  
Ans. Here, we have to use Self-Join as the requirement wants us to analyze the EmployeeDetails table as two different tables, each for Employee and manager records.

**SELECT DISTINCT** E.FullName

**FROM** EmpDetails E

**INNER** **JOIN** EmpDetails M

**ON** E.EmpID = M.ManagerID;

**Ques.7. Write a SQL query to fetch all employee records from EmployeeDetails table who have a salary record in EmployeeSalary table.**  
Ans. Using 'Exists'-

**SELECT** \* **FROM** EmployeeDetails E

**WHERE** **EXISTS**

(**SELECT** \* **FROM** EmployeeSalary S **WHERE** E.EmpId = S.EmpId);

**Ques.8. Write a SQL query to fetch duplicate records from a table.**  
Ans. In order to find duplicate records from table we can use GROUP BY on all the fields and then use HAVING clause to return only those fields whose count is greater than 1 i.e. the rows having duplicate records.

**SELECT** EmpId, Project, Salary, **COUNT**(\*)

**FROM** EmployeeSalary

**GROUP** **BY** EmpId, Project, Salary

**HAVING** **COUNT**(\*) > 1;

**Ques.9. Write a SQL query to remove duplicates from a table without using temporary table.**  
Ans. Using Group By and Having clause-

**DELETE** **FROM** EmployeeSalary

**WHERE** EmpId **IN** (

**SELECT** EmpId

**FROM** EmployeeSalary

**GROUP** **BY** Project, Salary

**HAVING** **COUNT**(\*) > 1));

Using rowId in Oracle-

**DELETE** **FROM** EmployeeSalary

**WHERE** rowid **NOT** **IN**

(**SELECT** **MAX**(rowid) **FROM** EmployeeSalary **GROUP** **BY** EmpId);

**Ques.10. Write a SQL query to fetch only odd rows from table.**  
Ans. This can be achieved by using Row\_number in SQL server-

**SELECT** E.EmpId, E.Project, E.Salary

**FROM** (

**SELECT** \*, Row\_Number() OVER(**ORDER** **BY** EmpId) **AS** RowNumber

**FROM** EmployeeSalary

) E

**WHERE** E.RowNumber % 2 = 1

**Ques.11. Write a SQL query to fetch only even rows from table.**  
Ans. Using the same Row\_Number() and checking that the remainder when divided by 2 is 0-

**SELECT** E.EmpId, E.Project, E.Salary

**FROM** (

**SELECT** \*, Row\_Number() OVER(**ORDER** **BY** EmpId) **AS** RowNumber

**FROM** EmployeeSalary

) E

**WHERE** E.RowNumber % 2 = 0

**Ques.12. Write a SQL query to create a new table with data and structure copied from another table.**  
Ans. Using SELECT INTO command-

**SELECT** \* **INTO** newTable **FROM** EmployeeDetails;

**Ques.13. Write a SQL query to create an empty table with same structure as some other table.**  
Ans. Using SELECT INTO command with False 'WHERE' condition-

**SELECT** \* **INTO** newTable **FROM** EmployeeDetails **WHERE** 1 = 0;

This can also done using mySQL 'Like' command with CREATE statement-

**CREATE** **TABLE** newTable **LIKE** EmployeeDetails;

**Ques.14. Write a SQL query to fetch common records between two tables.**  
Ans. Using INTERSECT-

**SELECT** \* **FROM** EmployeeSalary

**INTERSECT**

**SELECT** \* **FROM** ManagerSalary

**Ques.15. Write a SQL query to fetch records that are present in one table but not in another table.**  
Ans. Using MINUS-

**SELECT** \* **FROM** EmployeeSalary

**MINUS**

**SELECT** \* **FROM** ManagerSalary

**Ques.16. Write a SQL query to find current date-time.**  
Ans. mySQL-

**SELECT** NOW();

SQL Server-

**SELECT** getdate();

Oracle-

**SELECT** SYSDATE **FROM** DUAL;

**Ques.17. Write a SQL query to fetch all the Employees details from EmployeeDetails table who joined in Year 2016.**  
Ans. Using BETWEEN for the date range '01-01-2016' AND '31-12-2016'-

**SELECT** \* **FROM** EmployeeDetails

**WHERE** DateOfJoining **BETWEEN** '01-01-2016' **AND** date '31-12-2016';

Also, we can extract year part from the joining date (using YEAR in mySQL)-

**SELECT** \* **FROM** EmployeeDetails

**WHERE** **YEAR**(DateOfJoining) = '2016';

**Ques.18. Write a SQL query to fetch top n records?**  
Ans. In mySQL using LIMIT-

**SELECT** \* **FROM** EmployeeSalary **ORDER** **BY** Salary **DESC** **LIMIT** N

In SQL server using TOP command-

**SELECT** TOP N \* **FROM** EmployeeSalary **ORDER** **BY** Salary **DESC**

In Oracle using ROWNUM-

**SELECT** \* **FROM** (**SELECT** \* **FROM** EmployeeSalary **ORDER** **BY** Salary **DESC**)

**WHERE** ROWNUM <= 3;

**Ques.19. Write SQL query to find the nth highest salary from table.**  
Ans. Using Top keyword (SQL Server)-

**SELECT** TOP 1 Salary

**FROM** (

**SELECT** **DISTINCT** TOP N Salary

**FROM** Employee

**ORDER** **BY** Salary **DESC**

)

**ORDER** **BY** Salary **ASC**

Using limit clause(mySQL)-

**SELECT** Salary **FROM** Employee **ORDER** **BY** Salary **DESC** **LIMIT** N-1,1;

**Ques.20. Write SQL query to find the 3rd highest salary from table without using TOP/limit keyword.**  
Ans. The below SQL query make use of correlated subquery wherein in order to find the 3rd highest salary the inner query will return the count of till we find that there are two rows that salary greater than other distinct salaries.

**SELECT** Salary

**FROM** EmployeeSalary Emp1

**WHERE** 2 = (

**SELECT** **COUNT**( **DISTINCT** ( Emp2.Salary ) )

**FROM** EmployeeSalary Emp2

**WHERE** Emp2.Salary > Emp1.Salary

)

For nth highest salary-

**SELECT** Salary

**FROM** EmployeeSalary Emp1

**WHERE** N-1 = (

**SELECT** **COUNT**( **DISTINCT** ( Emp2.Salary ) )

**FROM** EmployeeSalary Emp2

**WHERE** Emp2.Salary > Emp1.Salary

)