100 SQL Practice Problems

Basic Queries (1-20)

1. Write a query to select all columns from a table `Employees`.
2. Write a query to select specific columns (`FirstName`, `LastName`) from `Employees`.
3. Write a query to select distinct values from the `Department` column.
4. Write a query to filter records where `Salary` is greater than 50000.
5. Write a query to filter records where `Department` is 'IT'.
6. Write a query to sort records by `LastName` in ascending order.
7. Write a query to sort records by `Salary` in descending order.
8. Write a query to select top 5 highest paid employees.
9. Write a query to find employees whose name starts with 'A'.
10. Write a query to find employees whose name ends with 'n'.
11. Write a query to find employees whose name contains 'an'.
12. Write a query to check for NULL values in `ManagerID`.
13. Write a query to check for NOT NULL values in `Email`.
14. Write a query to use the `IN` operator to find employees in 'IT' or 'HR'.
15. Write a query to use the `BETWEEN` operator to find salaries between 30000 and 50000.
16. Write a query to use the `LIKE` operator with wildcards.
17. Write a query to use the `AND` operator.
18. Write a query to use the `OR` operator.
19. Write a query to use the `NOT` operator.
20. Write a query to alias columns in the result set.

Aggregate Functions (21-35)

21. Write a query to count the total number of employees.
22. Write a query to calculate the total salary of all employees.
23. Write a query to calculate the average salary.
24. Write a query to find the minimum salary.
25. Write a query to find the maximum salary.
26. Write a query to count the number of distinct departments.
27. Write a query to group employees by `Department` and count them.
28. Write a query to group employees by `Department` and find average salary.
29. Write a query to filter groups using `HAVING` (e.g., departments with > 5 employees).
30. Write a query to find the department with the highest average salary.
31. Write a query to find the department with the lowest total salary.
32. Write a query to calculate the standard deviation of salaries.
33. Write a query to calculate the variance of salaries.
34. Write a query to find the sum of salaries for each job title.
35. Write a query to count employees hired in each year.

Joins (36-55)

36. Write a query to perform an INNER JOIN between `Employees` and `Departments`.
37. Write a query to perform a LEFT JOIN between `Employees` and `Departments`.
38. Write a query to perform a RIGHT JOIN between `Employees` and `Departments`.
39. Write a query to perform a FULL OUTER JOIN.
40. Write a query to perform a CROSS JOIN.
41. Write a query to perform a SELF JOIN to find managers of employees.
42. Write a query to find employees who do not belong to any department.
43. Write a query to find departments that have no employees.
44. Write a query to join three tables: `Employees`, `Departments`, and `Locations`.
45. Write a query to find employees working in 'New York'.
46. Write a query to find the manager's name for each employee.
47. Write a query to find employees who earn more than their managers.
48. Write a query to find employees who have the same job title as their manager.
49. Write a query to list all employees and their projects (Many-to-Many).
50. Write a query to find projects with no employees assigned.
51. Write a query to find employees assigned to multiple projects.
52. Write a query to find the department name and location for each employee.
53. Write a query to find the total salary expense for each location.
54. Write a query to list all possible combinations of employees and projects.
55. Write a query to find pairs of employees who live in the same city.

Subqueries (56-70)

56. Write a query to find employees with a salary greater than the average salary.
57. Write a query to find employees who work in the 'IT' department using a subquery.
58. Write a query to find the employee with the second highest salary.
59. Write a query to find employees who earn more than the maximum salary in 'HR'.
60. Write a query to find departments with more than the average number of employees.
61. Write a query to using a correlated subquery.
62. Write a query to check for existence using `EXISTS`.
63. Write a query to check for non-existence using `NOT EXISTS`.
64. Write a query to find employees who have not been assigned a project.
65. Write a query to find the department with the most employees.
66. Write a query to find the highest paid employee in each department.
67. Write a query to delete duplicate rows using a subquery.

68. Write a query to update salaries based on department average.
69. Write a query to find the 3rd highest salary without using `LIMIT`.
70. Write a query to select the last record from a table.

Advanced & DDL/DML (71-100)

71. Write a query to create a new table `Customers`.
72. Write a query to insert a new record into `Customers`.
73. Write a query to update an existing record.
74. Write a query to delete a record.
75. Write a query to truncate a table.
76. Write a query to drop a table.
77. Write a query to add a column to an existing table.
78. Write a query to drop a column from a table.
79. Write a query to rename a column.
80. Write a query to create a primary key constraint.
81. Write a query to create a foreign key constraint.
82. Write a query to create a unique constraint.
83. Write a query to create a check constraint.
84. Write a query to create a default constraint.
85. Write a query to create an index on a column.
86. Write a query to create a view.
87. Write a query to select data from a view.
88. Write a query to drop a view.
89. Write a query to create a stored procedure.
90. Write a query to execute a stored procedure.
91. Write a query to create a trigger.
92. Write a query to use `CASE` statement.
93. Write a query to use `COALESCE` function.
94. Write a query to use `NULLIF` function.
95. Write a query to use string functions (`CONCAT`, `SUBSTRING`, `UPPER`, `LOWER`).
96. Write a query to use date functions (`NOW`, `DATE_ADD`, `DATEDIFF`).
97. Write a query to use window functions (`ROW_NUMBER`, `RANK`, `DENSE_RANK`).
98. Write a query to calculate running total.
99. Write a query to find gaps in a sequence of numbers.
100. Write a query to pivot data (rows to columns).