

SQL Cheat Sheet: Intermediate - LIKE, ORDER BY, GROUP BY

	Command	Syntax	Description	Example
LIKE		SELECT column1, column2, ... FROM table_name WHERE column1 LIKE pattern;	LIKE operator is used in a WHERE clause to search for a specified pattern in a column. Two wildcards often used in conjunction with the LIKE operator are percent sign(%) and underscore sign (_), depending upon the SQL engine being used. The asterisk operator selects values within a given range. The values can be numbers, text, or dates. The across operator is inclusive: begin and end values are included. order by keyword is used to sort the result set in ascending or descending order. The default is ascending. In case of multiple columns in ORDER BY, the sorting will be done in the sequence of the appearance of the arguments. order by clause is used in collaboration with the SELECT statement to arrange data with identical values into groups. having clause is used in conjunction with order by clause in collaboration with the SELECT statement in order to filter the data as per the given condition and then group as per identical values of a specified parameter.	SELECT * name , 1, name FROM employees WHERE address LIKE '%12345, IL'; This command will output all entries with zip=1, in the Address. SELECT * from employees where salary between 40000 and 80000; This generates all records of employees with salaries between 40000 and 80000. SELECT * name, 1, name, dept as first_department FROM emp as emp, 1, name; This displays the first name, last name, and department ID of employees, first sorted in descending order of department IDs and then sorted alphabetically as per their last names. SELECT emp.dept, count(*) FROM employees GROUP BY emp.dept; This returns the department IDs and the number of employees in them, grouped by the department IDs. SELECT emp.dept, count(*) FROM employees GROUP BY emp.dept; This returns the department IDs along with the number of employees in them, grouped by the department IDs.
BETWEEN		SELECT column1, column2, ... FROM table_name WHERE column_name BETWEEN value1 AND value2;		
ORDER BY		SELECT column1, column2, ... FROM table_name ORDER BY column1, column2, ... ASC DESC;		
GROUP BY		SELECT column1, column2, ... FROM table_name GROUP BY column_name(s);		
HAVING		SELECT column_name(s) FROM table_name GROUP BY column_name(s) HAVING condition		

Author(s)

[Lakshmi Holla](#)

[Abhinav Gupta](#)

[Bhaskar Gupta](#)



Changelog

Date	Version	Changed by	Change Description
2023-10-03	1.3	Steve Ford	QA pass with edits
2023-10-01	1.2	Abhinav Gupta	Updated the document
2023-05-04	1.1	Bunny Li	Formatting changes
2021-07-28	1.0	Lakshmi Holla	Initial Version