SQL Cheat Sheet

1. Basic SQL Queries

Task	SQL Command	Example
Select all columns	SELECT * FROM table_name;	SELECT * FROM employees;
Select specific columns	SELECT column1, column2 FROM table_name;	SELECT first_name, salary FROM employees;
Select distinct values	SELECT DISTINCT column_name FROM table_name;	SELECT DISTINCT department FROM employees;

2. Filtering Data

Task	SQL Command	Example
Use WHERE clause	SELECT * FROM table_name WHERE condition;	SELECT * FROM employees WHERE age > 30;
AND condition	SELECT * FROM table WHERE condition1 AND condition2;	SELECT * FROM employees WHERE age > 30 AND salary > 50000;
OR condition	SELECT * FROM table WHERE condition1 OR condition2;	SELECT * FROM employees WHERE department = 'HR' OR salary > 50000;
NOT condition	SELECT * FROM table WHERE NOT condition;	SELECT * FROM employees WHERE NOT department = 'HR';
IN operator	SELECT * FROM table WHERE column IN (value1, value2);	SELECT * FROM employees WHERE department IN ('HR', 'IT');
BETWEEN operator	SELECT * FROM table WHERE column BETWEEN value1 AND value2;	SELECT * FROM employees WHERE salary BETWEEN 40000 AND 70000;
LIKE operator	SELECT * FROM table WHERE column LIKE 'pattern';	SELECT * FROM employees WHERE first_name LIKE 'A%';
Wildcards (%, _)	% (matches any sequence of characters)	_ (matches a single character)
EXISTS condition	SELECT * FROM table WHERE EXISTS (subquery);	SELECT * FROM employees WHERE EXISTS (SELECT 1 FROM departments WHERE employees.dept_id = departments.id);

3. Sorting and Limiting Results

Task	SQL Command	Example
Order results ASC/DESC	`SELECT * FROM table ORDER BY column ASC	DESC;`
Select top rows (MySQL)	SELECT * FROM table LIMIT number;	SELECT * FROM employees LIMIT 5;
Select top rows (SQL Server)	SELECT TOP number * FROM table;	SELECT TOP 10 * FROM employees;

4. SQL Aggregate Functions

Function	SQL Command	Example
MIN	SELECT MIN(column) FROM table;	SELECT MIN(salary) FROM employees;
MAX	SELECT MAX(column) FROM table;	SELECT MAX(salary) FROM employees;
COUNT	SELECT COUNT(column) FROM table;	SELECT COUNT(*) FROM employees;
SUM	SELECT SUM(column) FROM table;	SELECT SUM(salary) FROM employees;
AVG	SELECT AVG(column) FROM table;	SELECT AVG(salary) FROM employees;

5. SQL Joins

Join Type	SQL Command	Example
Inner Join	SELECT * FROM table1 INNER JOIN table2 ON table1.id = table2.id;	SELECT employees.name, departments.department FROM employees INNER JOIN departments ON employees.dept_id = departments.id;
Left Join	SELECT * FROM table1 LEFT JOIN table2 ON table1.id = table2.id;	SELECT employees.name, departments.department FROM employees LEFT JOIN departments ON employees.dept_id = departments.id;
Right Join	SELECT * FROM table1 RIGHT JOIN table2 ON table1.id = table2.id;	SELECT employees.name, departments.department FROM employees RIGHT JOIN departments ON employees.dept_id = departments.id;
Full Join	SELECT * FROM table1 FULL JOIN table2 ON table1.id = table2.id;	SELECT employees.name, departments.department FROM employees FULL JOIN departments ON employees.dept_id = departments.id;
Self Join	SELECT a.name, b.name FROM table a, table b WHERE a.manager_id = b.id;	SELECT e1.name, e2.name FROM employees e1, employees e2 WHERE e1.manager_id = e2.id;
Union	SELECT column FROM table1 UNION SELECT column FROM table2;	SELECT name FROM employees UNION SELECT name FROM managers;

6. SQL Grouping and Filtering Groups

Task	SQL Command	Example
Group By clause	SELECT column, COUNT(*) FROM table GROUP BY column;	SELECT department, COUNT(*) FROM employees GROUP BY department;
Having clause	SELECT column, COUNT(*) FROM table GROUP BY column HAVING COUNT(*) > value;	SELECT department, COUNT(*) FROM employees GROUP BY department HAVING COUNT(*) > 10;

7. Inserting, Updating, and Deleting Data

Task	SQL Command	Example
Insert data	INSERT INTO table (column1, column2) VALUES (value1, value2);	INSERT INTO employees (name, salary) VALUES ('John Doe', 50000);
Insert from another table	INSERT INTO table1 SELECT * FROM table2;	INSERT INTO employees_backup SELECT * FROM employees;
Update data	UPDATE table SET column=value WHERE condition;	UPDATE employees SET salary = 60000 WHERE id = 1;
Delete data	DELETE FROM table WHERE condition;	DELETE FROM employees WHERE id = 1;

8. SQL Case Statements

Task	SQL Command	Example
Case statement	SELECT column, CASE WHEN condition THEN result ELSE result END FROM table;	SELECT name, CASE WHEN salary > 50000 THEN 'High' ELSE 'Low' END AS SalaryCategory FROM employees;

9. SQL Stored Procedures

Task	SQL Command	Example
Create stored procedure	CREATE PROCEDURE procedure_name AS SQL_statement;	CREATE PROCEDURE GetAllEmployees AS SELECT * FROM employees;
Execute stored procedure	EXEC procedure_name;	EXEC GetAllEmployees;

10. SQL Operators

Operator	Description
=	Equal to
<> or !=	Not equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
BETWEEN	Within a range
IN	Matches values in a list
LIKE	Matches a pattern