SQL Zero to Hero Cheat Sheet

1. SQL Introduction

SQL (Structured Query Language) is the standard language used to store, retrieve, and manipulate data in Relational Database Management Systems (RDBMS) like MySQL, PostgreSQL, SQL Server, and Oracle.

2. SQL Categories

- DDL: Define structure (e.g., CREATE TABLE)
- DML: Manipulate data (e.g., INSERT, UPDATE, DELETE)
- DQL: Query data (e.g., SELECT)
- DCL: Manage access (e.g., GRANT, REVOKE)
- TCL: Handle transactions (e.g., COMMIT, ROLLBACK)

3. Basic SQL Syntax

SELECT name, age FROM students;
SELECT * FROM students WHERE age > 18;
SELECT DISTINCT course FROM students;
SELECT * FROM students WHERE age > 18 AND course = 'Math';
SELECT * FROM students ORDER BY age DESC;
SELECT * FROM students LIMIT 5 OFFSET 10;

4. Functions

Aggregate: COUNT(), SUM(), AVG(), MIN(), MAX()
String: UPPER(name), LENGTH(name), CONCAT()
Date: NOW(), CURDATE(), DATEDIFF()
Example: SELECT COUNT(*), AVG(salary) FROM employees;

5. Joins

INNER JOIN: Matching rows in both tables
LEFT JOIN: All from left + matched from right
RIGHT JOIN: All from right + matched from left
FULL JOIN: All rows from both tables
Example: SELECT orders.id, customers.name FROM orders JOIN customers ON orders.customer_id = customers.id;

6. Grouping & Filtering

GROUP BY: SELECT dept, COUNT(*) FROM employees GROUP BY dept;
HAVING: SELECT dept, COUNT(*) FROM employees GROUP BY dept HAVING COUNT(*) > 2;

7. Subqueries

Scalar: Returns a single value
Correlated: Depends on outer query
Example: SELECT name FROM emp WHERE salary > (SELECT AVG(salary) FROM emp);

SQL Zero to Hero Cheat Sheet

8. CTE

WITH high_salary AS (SELECT * FROM employees WHERE salary > 10000) SELECT name FROM high_salary;

9. DDL

CREATE TABLE users (id INT, name VARCHAR(100));
ALTER TABLE users ADD age INT;
DROP TABLE users;

10. DML

INSERT INTO users VALUES (1, 'Alice');
UPDATE users SET name = 'Bob' WHERE id = 1;
DELETE FROM users WHERE id = 1;

11. DCL

GRANT SELECT ON students TO 'user1';
REVOKE INSERT ON students FROM 'user1';

12. TCL

BEGIN; UPDATE...; COMMIT; ROLLBACK; SAVEPOINT;

13. Constraints

NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK, DEFAULT

14. Indexes

CREATE INDEX idx_name ON users(name); DROP INDEX idx_name;

15. Views

CREATE VIEW active_users AS SELECT * FROM users WHERE is_active = 1;

16. Advanced SQL

CASE, COALESCE, EXISTS, UNION

17. Normalization

1NF: Atomic values
2NF: No partial dependency
3NF: No transitive dependency
BCNF: Determinant must be a candidate key